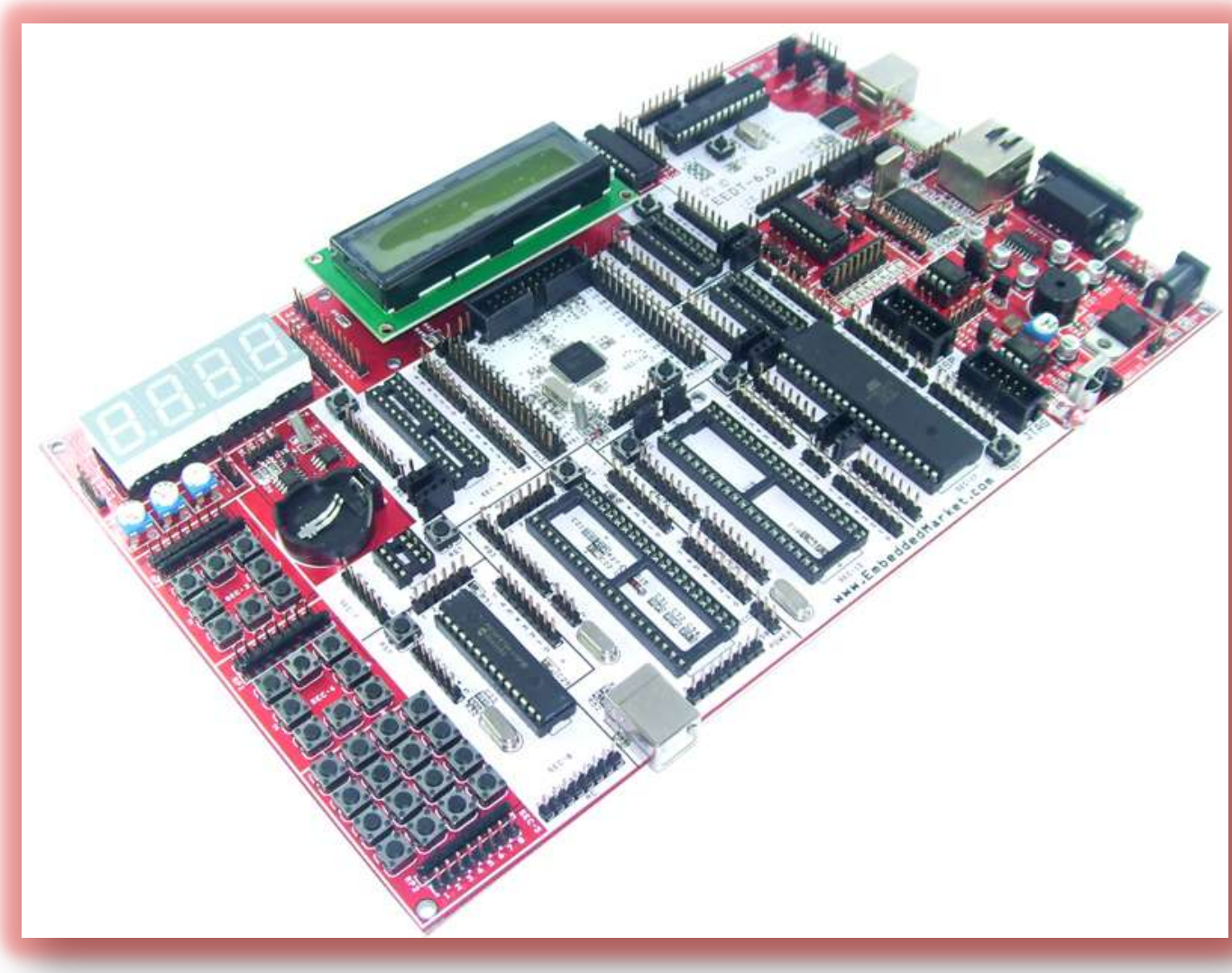
Embedded Engineer's Development Tool 6.0



First time in the world,

One development board for

Five Microcontroller Families

AVR + 8051 + PIC + ARM + ARDUINO

with Programmers

Sr. Num.	Topics	Page
1	About EEDT 6.0	3
2	Microcontrollers included in EEDT6.0 product pack.	5
3	Microcontrollers supported by EEDT6.0.	6
4	Programmers included in EEDT6.0 product pack	7
5	List of accessories included in EEDT6.0 product pack.	8
6	Interface circuits mounted on EEDT6.0 board	9
7	Programming onboard AVR microcontrollers	10
8	Programming onboard P89V51RD2 & ARM microcontroller	14
9	Programming onboard PIC microcontrollers	18
10	Using onboard Arduino	19
11	Using EEDT6.0 hardware board	20
12	Important information	36

1. About EEDT 6.0

Embedded Engineer's Development Tool (EEDT) is our oldest product offering.

It has evolved over last 5 years. Initially EEDT was designed for AVR and 8051 microcontrollers.

Later the PIC microcontrollers were supported.

Now EEDT6.0 completes most common microcontroller families by offering support to ARM and Arduino in addition to classic 8051, AVR and PIC microcontrollers.

Arduino cannot be exactly classified as the Microcontroller Family. Arduino is most versatile Open-Source Physical Computing platform based on AVR microcontrollers.

Since beginning of our EEDT product range, our emphasis was always on offering multiple microcontroller support on single development board with variety of famous interface circuits. EEDT has always been a complete set of development system. ISP Programmers, IDEs, Compilers, Sample Source Code, Cables, Connectors and power supply were part of the standard packing. In export shipment, we do not include certain low cost – high weight accessories like cables. Power supplies are also not included as every country has unique wall mounting sockets.

The EEDT 6.0 Development board includes following sections on the Board:

(Below sectional numbers are marked on the Hardware Board)

Sec -1] Seven Segment Display - Four multiplexed

Sec–2] Variable resistance for Analog Input – Three separate presets

Sec-3] Pulled down Push to On Switches – Eight Switches

Sec-4-] Pulled up Push to On Switches – Eight Switches

Sec-5] Matrix keypad of 4x4 keys – Total Sixteen Keys

Sec-6] I2C Based Real time clock, EEPROM & Digital to Analog Converter

Sec-7] Mount here ATTiny13 and other Pin compatible AVR Microcontrollers

Sec-8] Mount here PIC16F873A and other Pin compatible PIC Microcontrollers

Sec-9] Mount here ATmega8 / ATmega168 and other Pin compatible AVR Microcontrollers

Sec-10] Mount here PIC18F4550 / PIC18F452 and other Pin Compatible PIC Microcontrollers

Sec-11] 16x2 LCD Interface. LCD is mounted on the board. Same place can be used for 20x4 LCD

Sec-12] Pre-soldered ARM7 - LPC2138 microcontroller

Sec-13] Mount here P89V51RD2 (8051 family) microcontroller. This place can also be used for ATmega8515 AVR and its pin compatible microcontrollers. Reset pin Jumper position change required. Set it to Vcc when using 8051 family microcontroller. Set it to Gnd in case of AVR series.

Sec-14] High Current Driver based on ULN2803 – Use it to drive Stepper motors, seven segment Displays.

Sec-15] Mount here ATTiny2313 and other pin compatible AVR Microcontrollers

Sec-16] Mount here ATTiny26 and other pin compatible AVR Microcontrollers

Sec-17] Mount here ATmega16 / ATmega32 / ATmega8535 and other pin compatible AVR Microcontrollers

Sec-18] Implementation of Arduino Duemilanove platform

Sec-19] DC Motor driver using L293D

Sec-20] A bank of 8 LEDs

Sec-21] SPI EEPROM AT93C46

Sec-22] Collection of sensors and other interfaces – Temperature sensor, Light Sensor, Infrared (IR) Sensor, 38KHz IR Receiver, IR Transmitter, Buzzer

Sec-23] USB to TTL converter interface – This is used for Arduino as well as a standalone USB to TTL interface. Jumper settings required.

Sec-24] Micro SD Card – Can be interfaced with 3.3V I/Os or with 5V I/Os. Jumper settings required.

Sec-25] ENC28J60 based Ethernet Interface. Use it to create web enabled applications.

Sec-26] RS232 to TTL Converter interface.

Sec-27] A Power supply section which includes 5V and 3.3V regulation. Wall mounting DC adapter which can source 500mA or higher at 9VDC is suitable. Exceeding input voltage may heat the onboard Voltage regulator ICs.

2. Microcontrollers included in EEDT6.0 product pack.

The EEDT6.0 is multi microcontroller support development board.

Following Microcontrollers are Soldered / Mounted / Included with the EEDT6.0 Product:

1. ATmega32 – Mounted on the Board
2. P89V51RD2 – Included as spare
3. LPC2138 – Soldered on the Board
4. PIC16F873A – Mounted on the Board
5. Other microcontrollers are supported but are "Not" included in the EEDT6.0 product. Supported indicates that the microcontrollers can be used on the EEDT6.0 hardware and needs to be purchased separately.

3. Microcontrollers supported by EEDT6.0

EEDT6.0 supports variety of microcontrollers. The word supported means technically supported. These microcontrollers may not be included in the product pack.

8051:

AT89S51, AT89S52, P89V51RD2 (Programmer for AT89S51 and AT89S52 is not included)

AVR:

ATmega8, ATmega168, ATmega328, ATmega16, ATmega32, ATmega8515, ATmega8535, ATmega48, ATmega88, ATmega162, ATmega164, ATmega324, ATmega644, ATmega1284

PIC:

PIC16F873A and pin compatible microcontrollers

PIC18F452 and pin compatible microcontrollers

PIC18F4550 and pin compatible microcontrollers

ARM

LPC2138, LPC2148 (LPC2138 is soldered on the board)

Arduino

ATmega328 for Duemilanove version (Mounted on the board)

ATmega168 for Diecimila version

ATmega8 for NG version

4. Programmers included in EEDT6.0 product pack

EEDT6.0 includes following programmers.

All programmers are of type ISP / ICSP.

ISP stands for In-circuit Serial Programmer. (Does not mean it works on Serial Port.)

The ISP programmer allows you to program the AVR microcontrollers, keeping them in-circuit. Thus it eliminates the need of moving the microcontroller from development board to the programmer unit and back.

ICSP is another name given to the In-Circuit programming method.

1. **On-board USB AVR Programmer.**

   This is ISP Programmer for AVR which works via USB port.

   AVR microcontrollers are programmed and are automatically switched to the Run mode once programming gets over.

   The programmer is based on **AVRISP**.

   To know more about procedure to use the on-board programmer for AVR, read further topic titled "Programming onboard AVR microcontrollers"

2. **External USB Programmer for P89V51RD2 and LPC2138 (and for their compatible microcontrollers from NXP**

   The external (not on the board) programmer for P89V51RD2 and LPC2138 is USB controlled programmer. This is also meant for in-circuit programming.

   The programmer is based on **FlashMagic.**

   To know more about procedure to use the programmer for P89V51RD2 and LPC2138, read further topic titled "Programming onboard 8051 microcontrollers" & "Programming onboard ARM microcontrollers"

3. **External Serial port based programmer for PIC microcontrollers**

   The PIC programmer is Serial Port based ICSP type. This programmer is observed to be slower than expected and needs serial port on PC / Laptop. It does not work with the USB to Serial converters of any make.

   The programmer is based on **IC-Prog / PICPgm**.

   To know more about procedure to use the programmer for PIC, read further topic titled "Programming onboard PIC microcontrollers"

5. List of accessories included in EEDT6.0 product pack

The EEDT6.0 product pack includes following items:

(* Marked items are not included in Export Shipments)

Export Shipments are defined as those shipments which are delivered out of India)

1. **EEDT6.0 Hardware board**
2. **USB Programmer for P89V51RD2, LPC2138 and compatible microcontrollers made by NXP**
3. **Serial Programmer for PIC Microcontrollers**
4. **USB A-B Cable**
5. **LAN (Ethernet) Cross-over Cable**
6. **Pin conversion Attachment for AVR Programmer**
7. **Female to Female 6 pin cable for PIC Programmer**
8. **A set of Female to Female single pin Connectors each of 30cm length- 40 numbers**
9. **Spares – 4MHz & 16MHz Crystal – 1 Each**
10. **Spares – 22pf Disc Capacitor – 2 nos**
11. **Spares – P89V51RD2 Microcontroller**
12. **CD**
13. **Male to Female Serial Cable (*)**
14. **9V DC 500mA Wall mounting Adapter (*)**

6. Interface circuits mounted on EEDT6.0 board

Driver circuits, Input Circuits, Output circuits, Input-Output Combined circuits are called as Interface circuits.

In any embedded system / embedded product, there has to be at least one microcontroller and one interface circuit.

Below is the list of Interface circuits mounted on the EEDT6.0:

1. A bank of 8 LEDs
2. A set of Seven Segment Displays – Four Displays multiplexed
3. A bank of 8 Switches in Pulled-down
4. A bank of 8 Switches in Pulled-up
5. A bank of 8 Switches in Pulled-down
6. 4x4 Key Matrix
7. Real time Clock using DS1307 (I2C interface)
8. EEPROM 256Kbit using 24C256 (I2C interface)
9. Digital to Analog Converter 12bit resolution using MCP4725 (I2C interface)
10. Set of 3 variable resistance (Preset) to use as Independent Analog Inputs
11. 16x2 LCD Display
12. USB to TTL converter using FT232R (Converts USB signal to 5V Serial interface signals)
13. Micro SD Card interface – Use it with 3.3V or 5V I/O interface lines
14. DC Motor Driver using L293D driver IC
15. Stepper Motor driver / High current driver using ULN2803
16. Ethernet Interface using ENC28J60 (SPI interface)
17. EEPROM AT93C46 - 1Kbyte (SPI Interface)
18. RS232 to TTL Interface using MAX232
19. Infrared LED Transmitter
20. Infrared Receiver with comparator (Digital Output)
21. Infrared Receiver for 38KHz Signal using TSOP1738
22. Temperature Sensor using LM35 (Analog Output)
23. Light Sensor using LDR (Analog Output)
24. Buzzer interface

Above interfaces are mounted on the EEDT6.0 hardware PCB.

These interfaces can be connected to any Microcontroller via Single pin connectors included with the product.

7. Programming onboard AVR microcontrollers

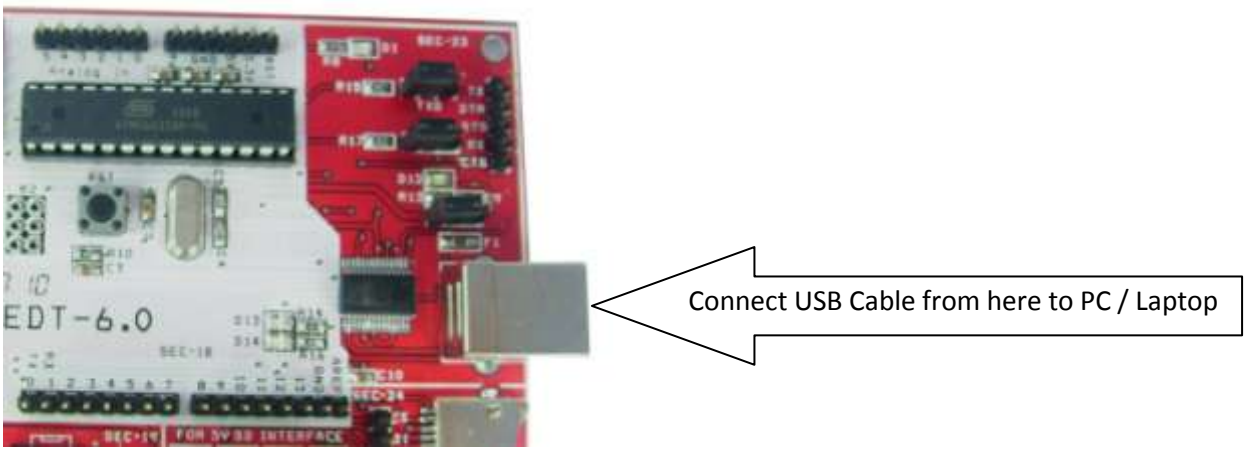
The Programmer for AVR Microcontroller is mounted on the EEDT6.0 Board.

This is based on Arduino.

The Arduino section on the EEDT6.0 can be used as Arduino or as the AVR Programmer.

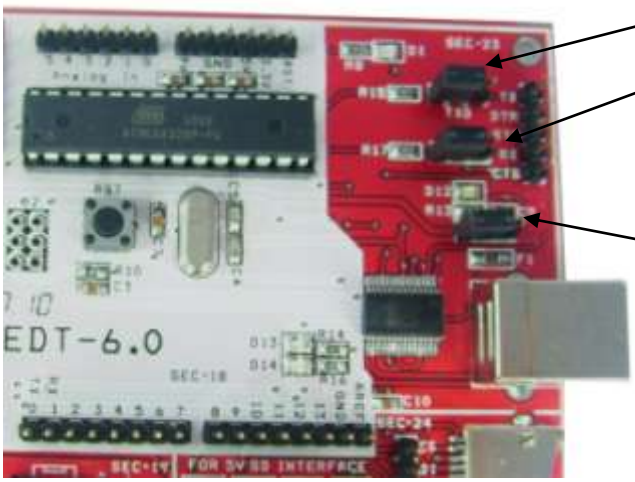
There are certain jumpers settings required to use this Section as Arduino or as the AVR Programmer

Do have a look at below enlarged sectional picture of EEDT6.0 showing the Arduino Section.



Steps for using Arduino as AVR Programmer: (Follow below settings marked as *(#)*)

Step 1 / 4: The Settings for USB to TTL converter for AVR Programmer:



USB to TTL For Arduino: (TXD & RXD)

(#)Insert to Left & Center Pin - Arduino can use the USB Port

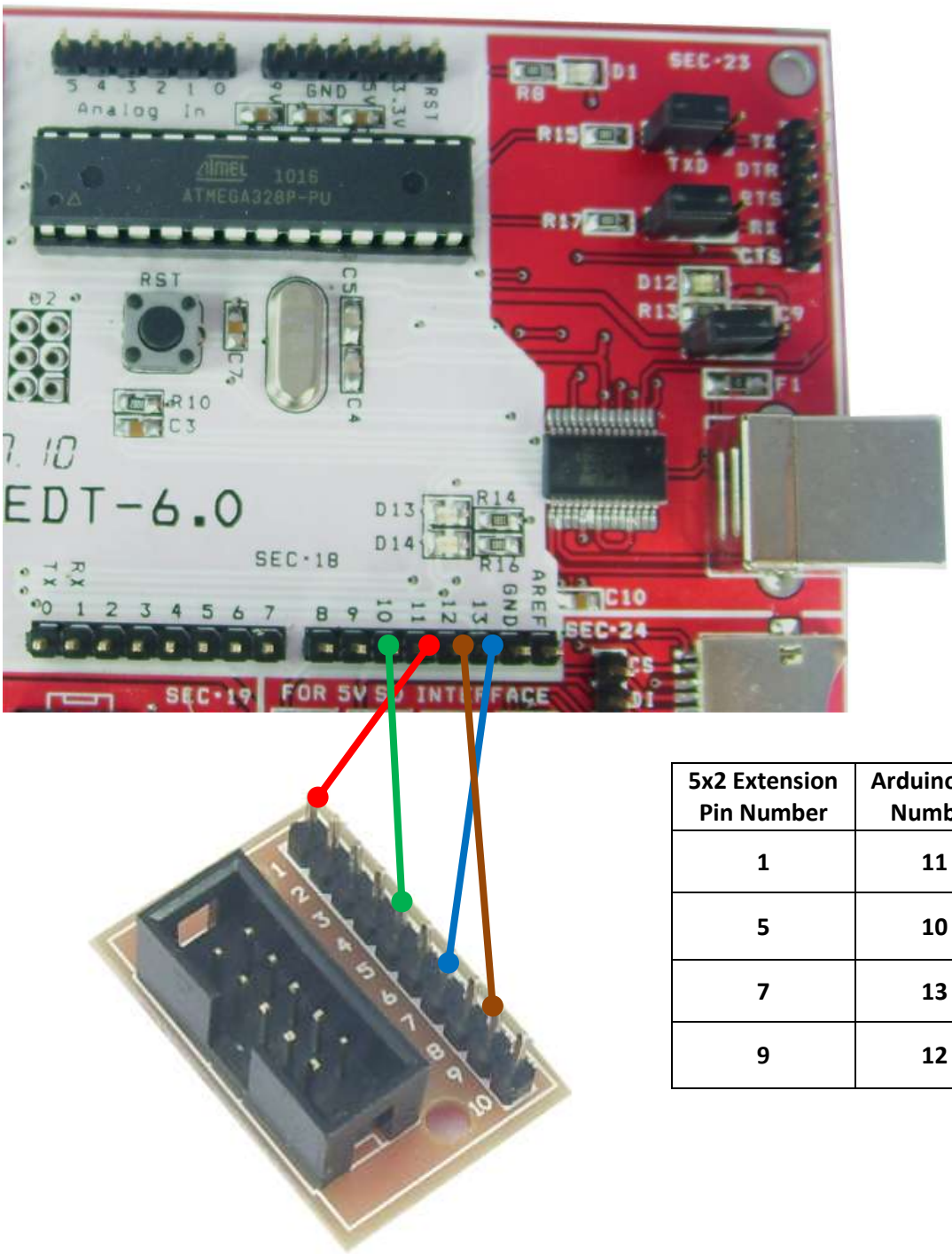
Insert to Right & Center Pin –USB to TTL is disconnected from Arduino and can be used as independent interface circuit.

Power Selector Jumper for Arduino Section:

(#)Insert to Left & Center Pin- 5V from onboard regulator.

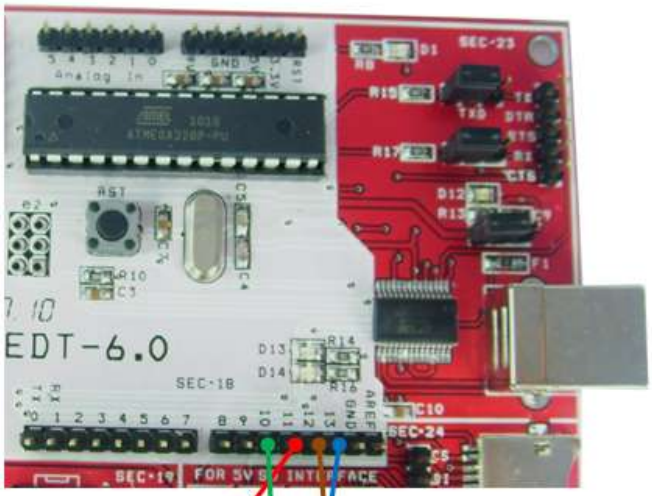
Insert to Right & Center Pin – 5V from USB Port

Step 2 / 4: **The 5x2 Extension board connections to the Arduino Section.:** ***(#)***

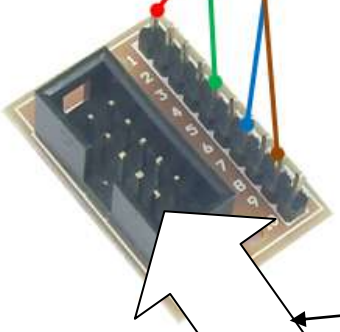


5x2 Extension Pin Number	Arduino Pin Number
1	11
5	10
7	13
9	12

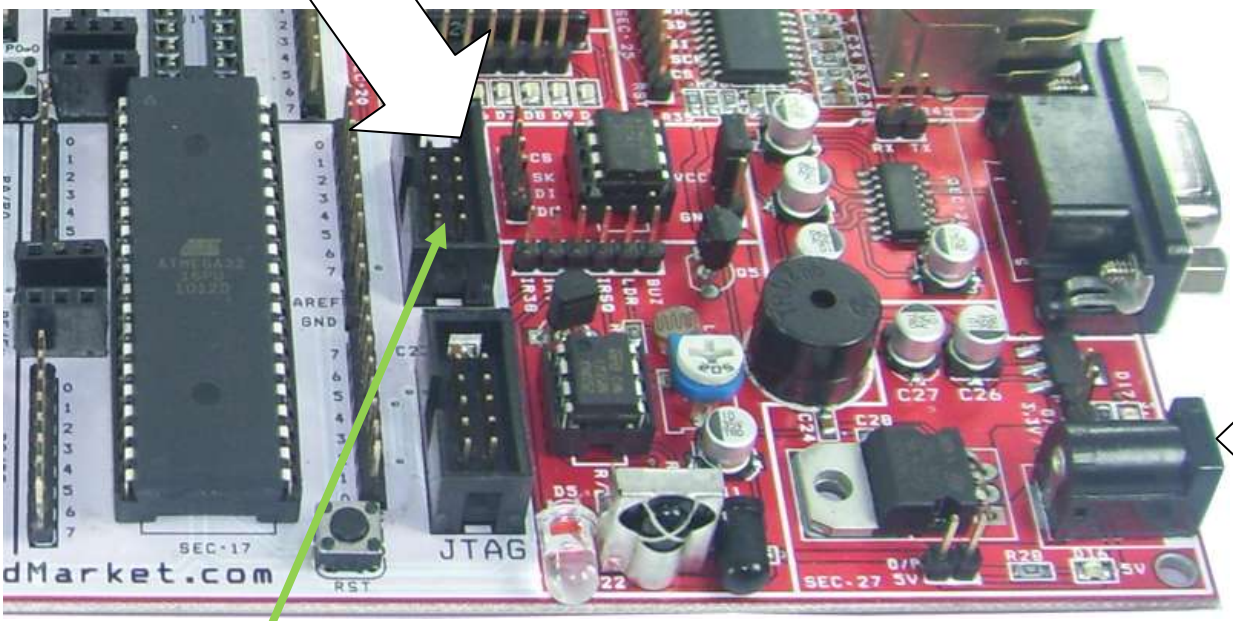
Step 3 / 4: **The 5x2 Extension board connections to ISP Port on EEDT6.0:** ***(#)***



5x2 Extension Pin Number	Arduino Pin Number
1	11
5	10
7	13
9	12



Connect FRC Cable between these two connectors.



Power up..

This ISP Socket is common to all AVR onboard microcontrollers / sockets for microcontrollers. In other words one ISP 5x2 port should be used for any AVR microcontroller that can be mounted on the board.

Step 4 / 4: Command line programming for AVR Programmer *(#)*

1. **Install WinAVR from the setup provided on the CD.**
2. **Start Command Prompt using Start => Run => Command Or Start => Accessories => Command Prompt**
3. **Change the path to working directory using CD DOS command**

   CD [Type Path of Your Project where the HEX file is located]

   Example:

   C:\Documents and Settings\User> CD E:\My Projects\LCDApp

   C:\Documents and Settings\User> E:

   This will change the directory to:

   E:\My Projects\LCDApp>

4. **Use further commands for programming Flash Memory**

   avrdude –p <controller-name> -c avrisp –b 19200 –P <comportnumber> -U flash:w:<hexfilename.hex>

   Use comportnumber=com1 to com8

   If your PC / Laptop configures the USB to Serial Interface beyond COM8 then edit the COM port number from Control Panel => Performance and Maintenance => System => Hardware Tab => Device Manager => Com Ports => Properties => Advance

Command for Program Flash hex:

avrdude –p m32 –c avrisp –b 19200 –P com6 –U flash:w:filename.hex:i

Command to Program HFUSE:

avrdude –p m32 –c avrisp –b 19200 –P com6 -U hfuse:w:fuse-value-in-hex-format:m

Command to Program LFUSE:

avrdude –p m32 –c avrisp –b 19200 –P com6 -U lfuse:w:fuse-value-in-hex-format:m

Command to Program EEPROM hex:

avrdude –p m32 –c avrisp –b 19200 –P com6 -U eeprom:w:filename.hex:m

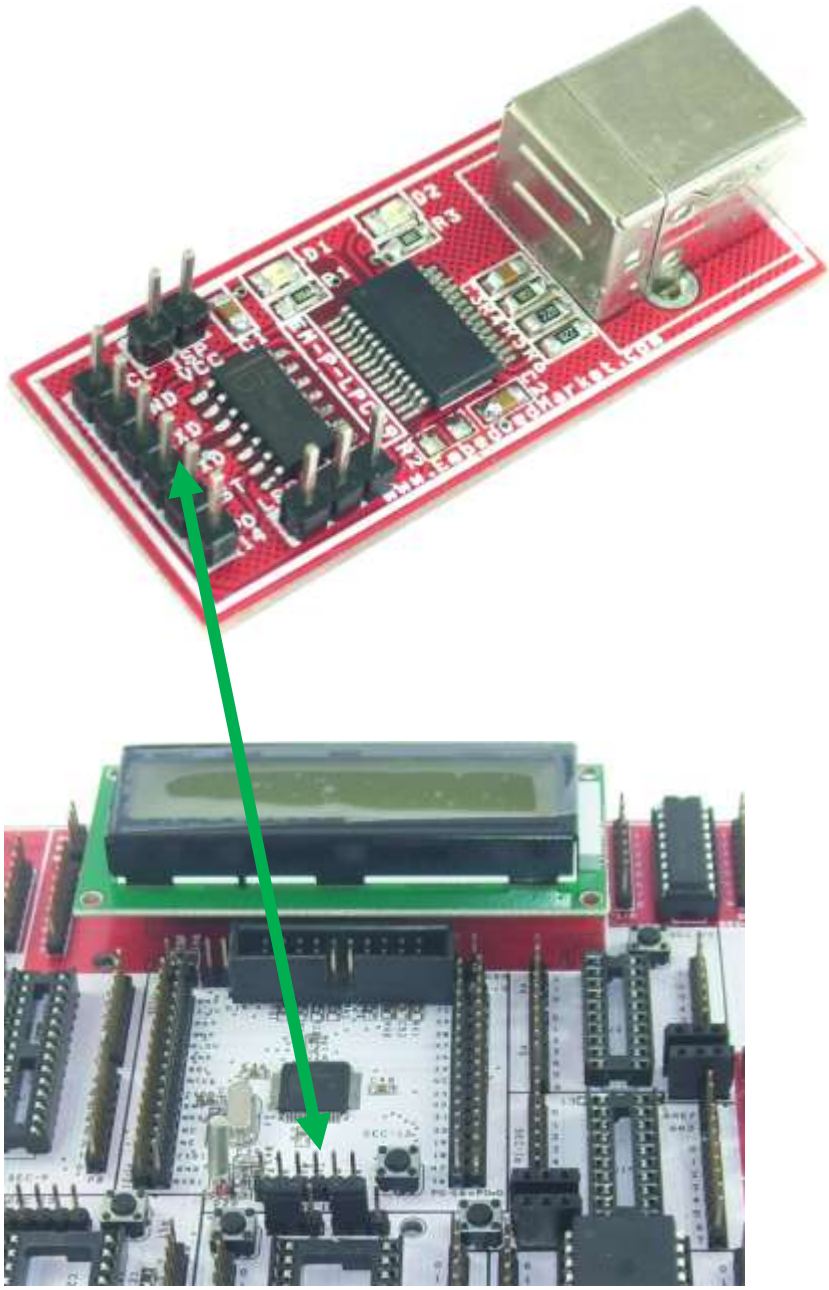
Command to Program Lock hex:

avrdude –p m32 –c avrisp –b 19200 –P com6 –U lock:w: fuse-value-in-hex-format:m

Not all above commands are required every time. The first command of Programming Flash is the most commonly required. This command will erase the existing program from ATmega32 and will re-program the hex file from filename.hex.

8. Programming onboard P89V51RD2 & ARM microcontroller

This picture shows the USB Programmer for NXP made P89V51 & ARM7 controllers provided along with the EEDT6.0

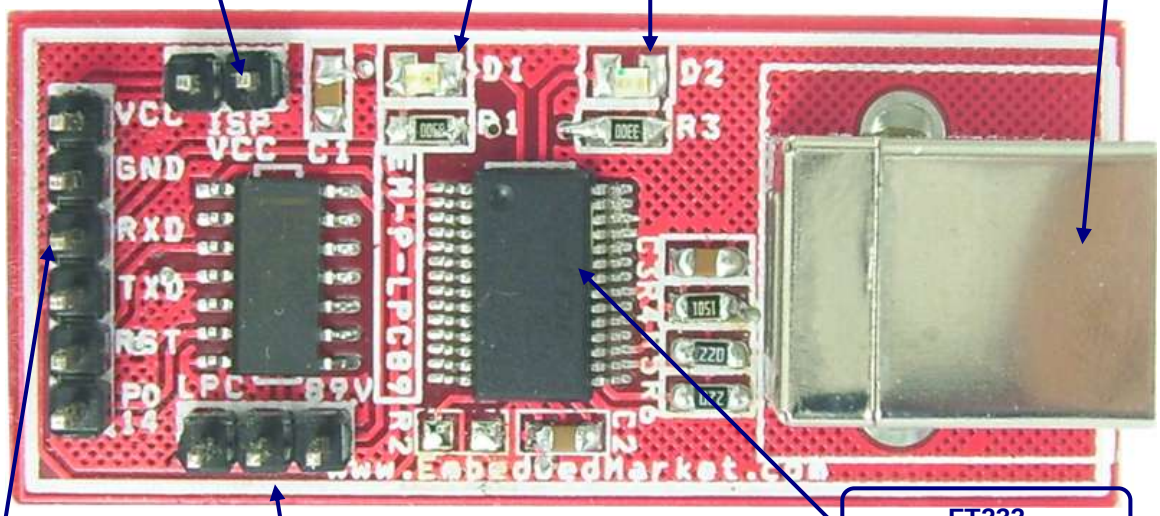


This hardware is designed to work with FlashMagic Programming Software. Connect 6 Pin Female to Female Cable as. shown for ARM Programming

Add Jumper here to connect 5V from USB to the development Board. Do not use this jumper with LPC ARM.

LEDs indicate program being transmitted from USB to the development board.

USB B Type



FT232

Selection Jumper:

For LPC ARM Programming: Add Jumper between Left pin and Center.

For P89V51RD2 Programming: Add Jumper between Right pin and Center.

Programming Header:

1. Vcc – 5V (Do not use for LPC ARM. If Jumper named ISP VCC is open then this pin Has No-Connections.)

2. GND- Common Ground between this programmer & the development board

3. RXD – Connect to RX line of the microcontroller. (RX0 in case of LPC ARM)

4. TXD – Connect to TX line of the microcontroller. (TX0 in case of LPC ARM)

5. RST – Connect to RESET line of the microcontroller

How to Program?

1. **Add / remove jumper as shown in above picture depending on the type of microcontroller being programmed.**

2. **Use the Female to Female connector provided with the product to connect the "Programming Header" to your development board.**

3. **Re-check the jumper selection again.**

4. **Connect A to B Type USB cable (also called as USB cable for printer) from the programmer to your PC / Laptop**

5. **If you have connected this Programmer for the first time then you may need to install the drivers. Download the drivers matching with the operating system installed on your PC / Laptop. The download link is : http://www.ftdichip.com/Drivers/VCP.htm**

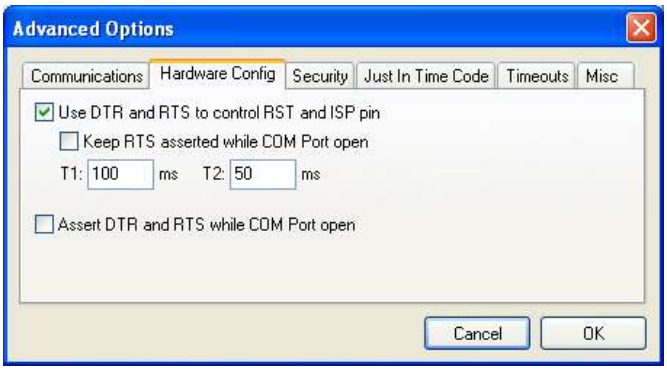
6. **After installing the drivers, you may download the FlashMagic Software from link: http://www.flashmagictool.com/**

7. **Install the FlashMagic software and start it from program menu.**

8. **In FlashMagic, Select the microcontroller's name, COM Port number, Baud Rate and other details as shown below:**

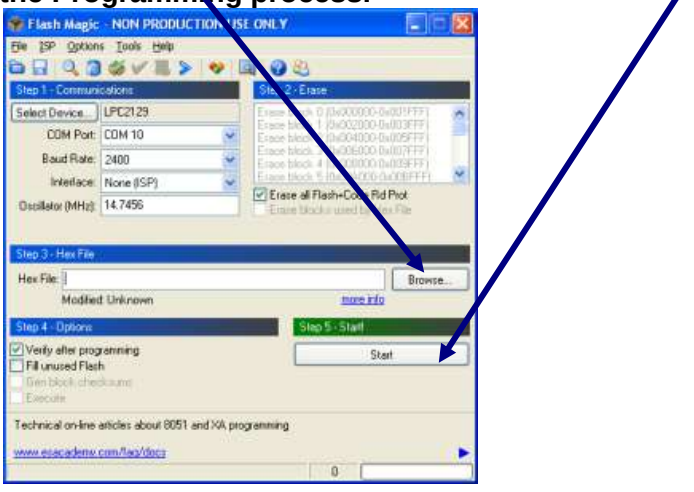
9. **In FlashMagic, click Options=>Advanced Options menu**

10. **It will open "Advanced Options". Make selection as shown below.**



11. **Click OK once done with the selection in "Advanced Options".**

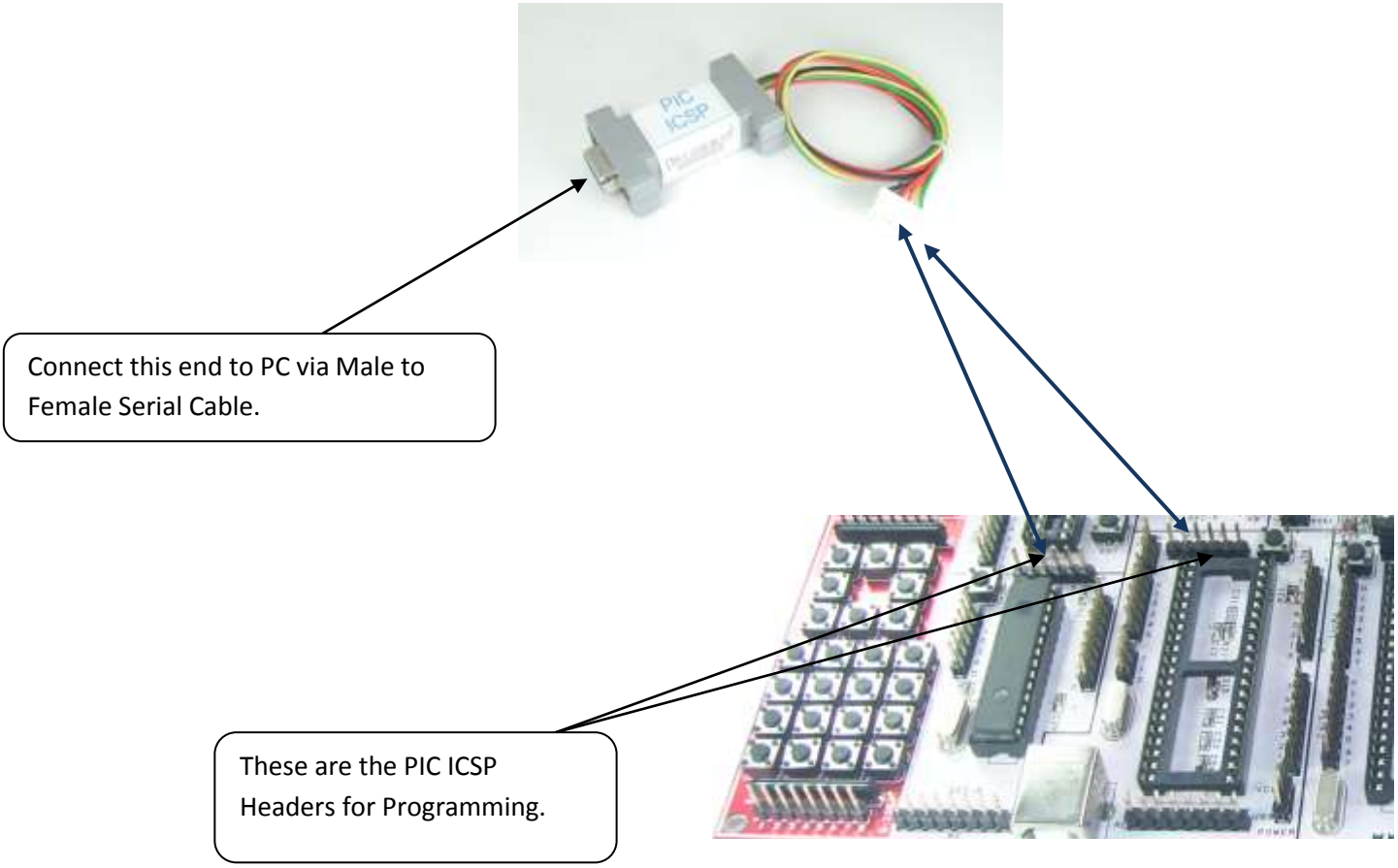
12. **Click "Browse" to select the hex file and click "Start" button in FlashMagic software to begin the Programming process.**



9. Programming onboard PIC microcontrollers

The PIC Programming hardware (as shown in the below picture) is designed to work with direct serial port (not USB to Serial converters).

There are several sofwtares which can be used along with this hardware for programming.



Use any of below listed Software to Program the PIC Microcontroller:

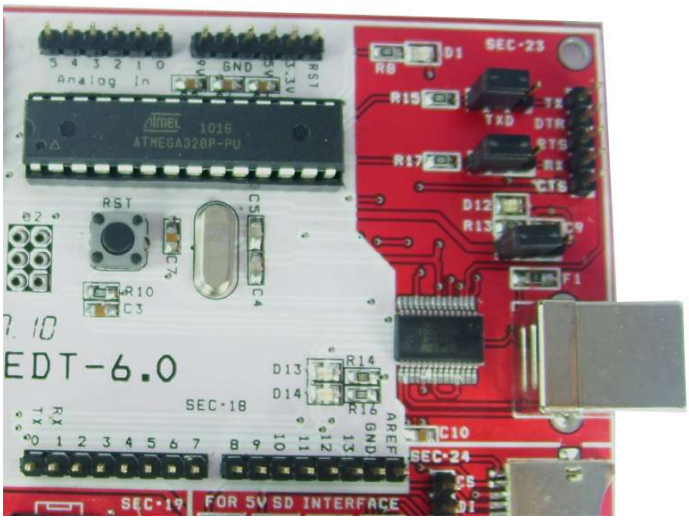
1. IC-Prog (http://www.ic-prog.com/icprog106B.zip)
2. PicPgm (Installable of this software is included in the CD)

10. Using onboard Arduino

This is Arduino compatible hardware (It is derived from Freeduino design).

Based on ATmega328 AVR controller.

In Chapter – 7, this Arduino hardware is used as AVR Programmer.



Thus the Arduino hardware is pre-loaded with the AVR programmer's application (Sketch in Arduino's standard terms).

The Sketch Credits to "Randall Bohn".

The Sketch (AVR Programmer's Arduino dependent Application) can be requested from support@embeddedmarket.com

This will allow you to load any other Arduino Sketch to the onbpard Arduino compatible hardware and re-laod the AVR Programmer's sketch when necessary.

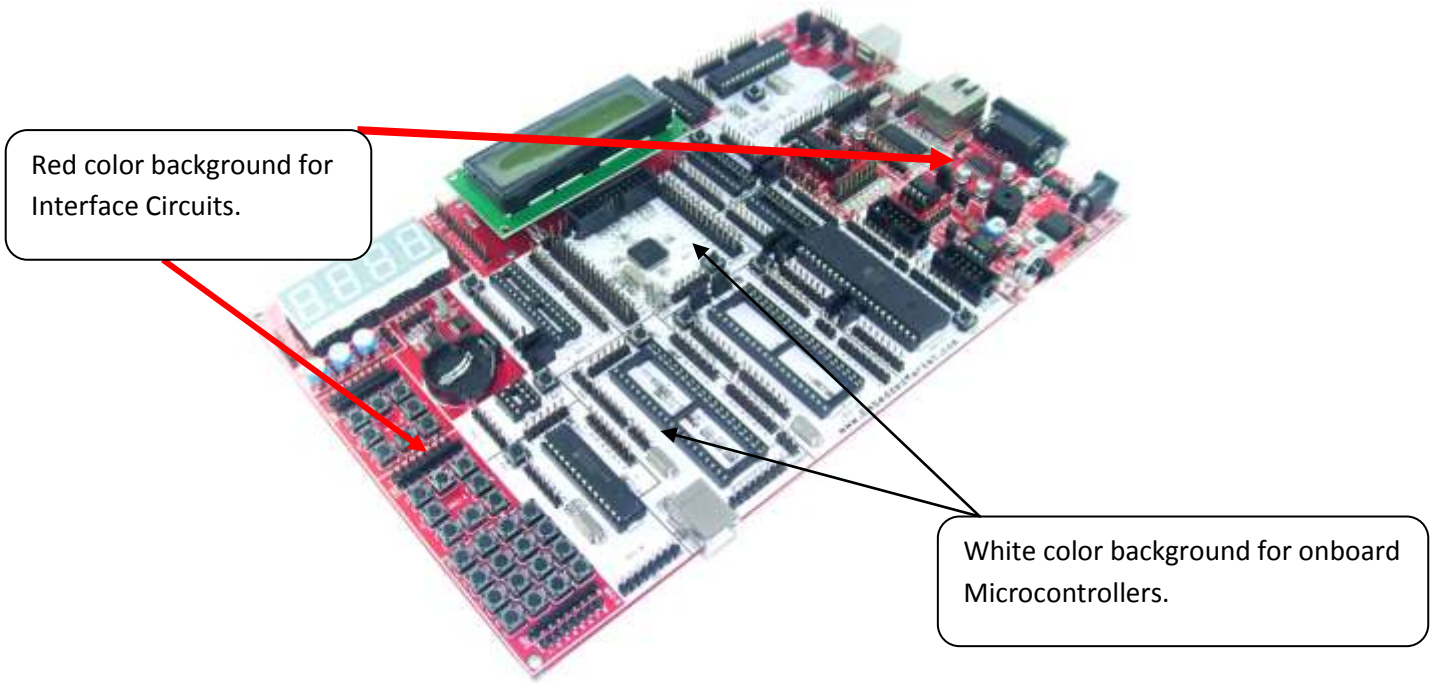
Arduino is Open Source Computing Platform.

Arduino simplifies program development and has extensive support at www.Arduino.cc

Arduino IDE setup is included in CD.

The Onboard Arduino Compatible Hardware is "Shield Friendly".

11. Using EEDT6.0 hardware board



The EEDT6.0 Hardware is divided in two major sections.

1. The onboard Microcontroller Section (This area has White color background on the Hardware board)
2. The onboard Interface Circuit Section (This area has Red color background on the Hardware board)

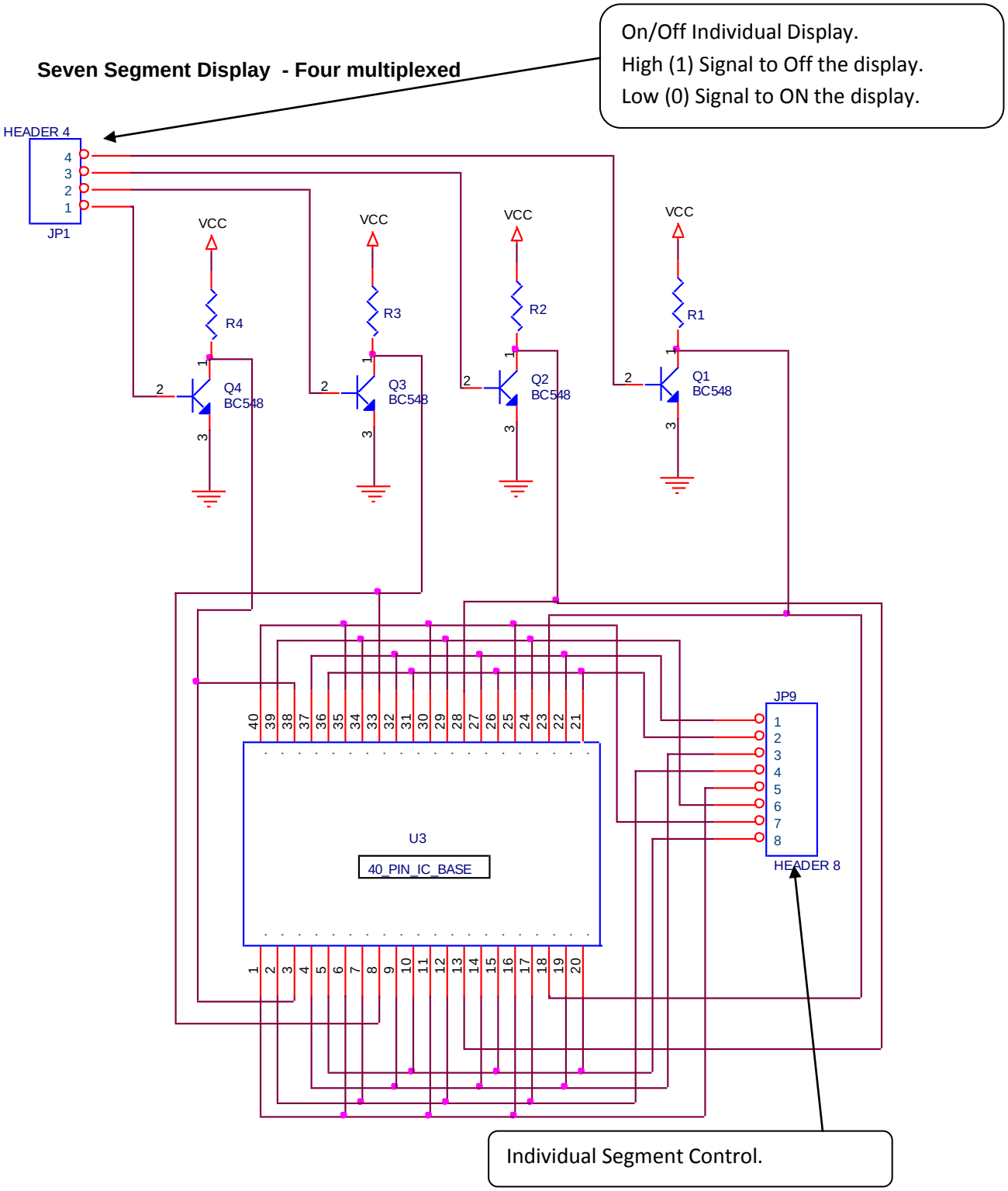
Input and Output pins of all Microcontroller circuits and Interface Circuits are open for connection.

This allows you to connect any interface to any onboard microcontroller using the single pin connectors as shown in the below picture. These connectors are included in the product pack.

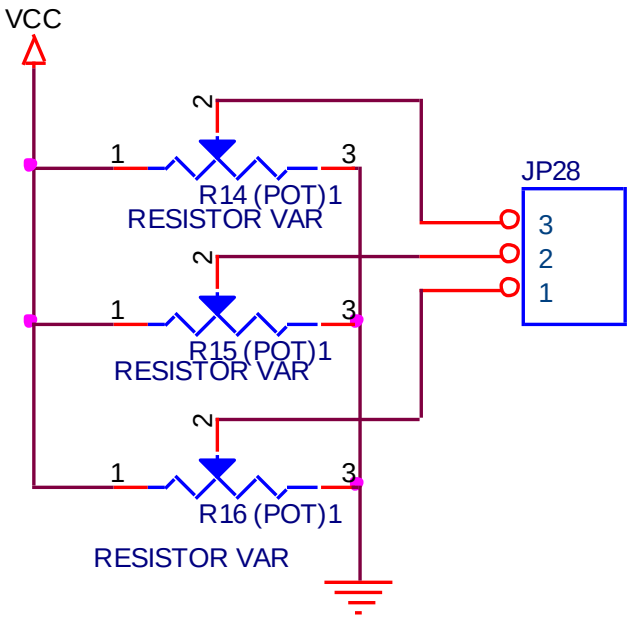
Required power (5V or 3.3V) to microcontrollers and interface circuits is already provided on the PCB. Thus you need not to provide the Vcc & Gnd lines to the microcontroller & interface sections.

Below are the sectional schematics. This will help you in connecting and interfacing microcontrollers.

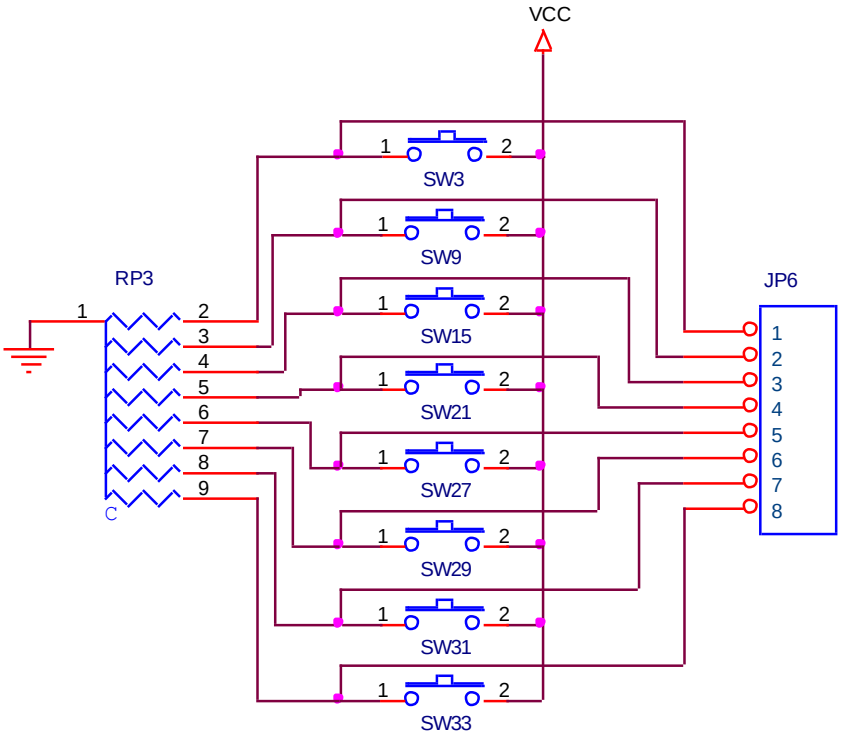
Sec -1] **Seven Segment Display - Four multiplexed**



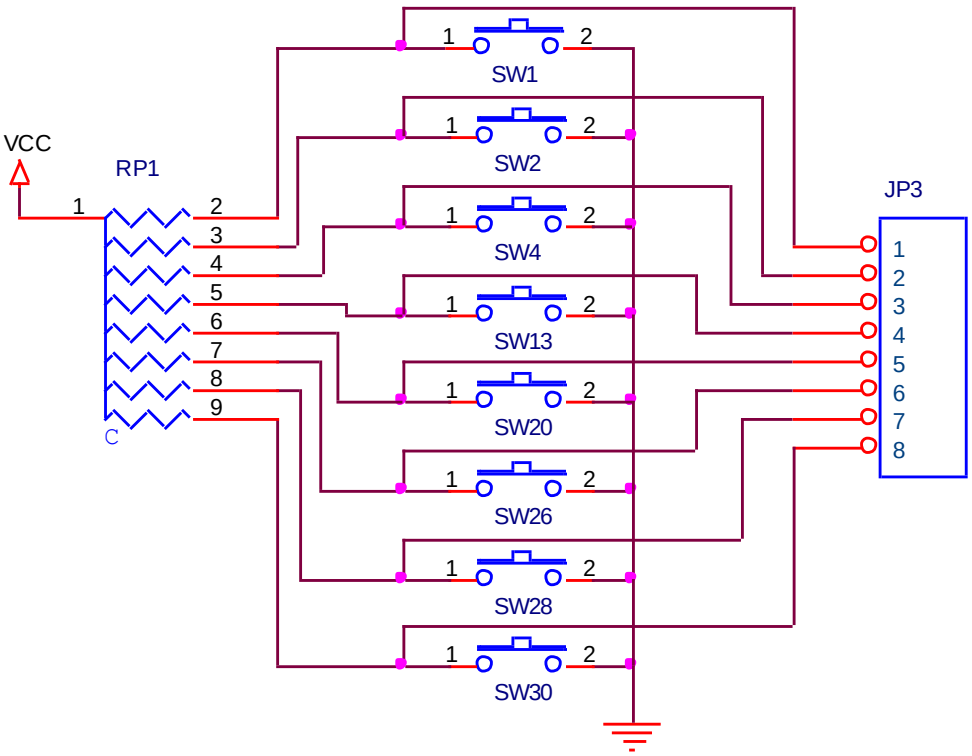
Sec–2] **Variable resistance for Analog Input – Three separate presets**



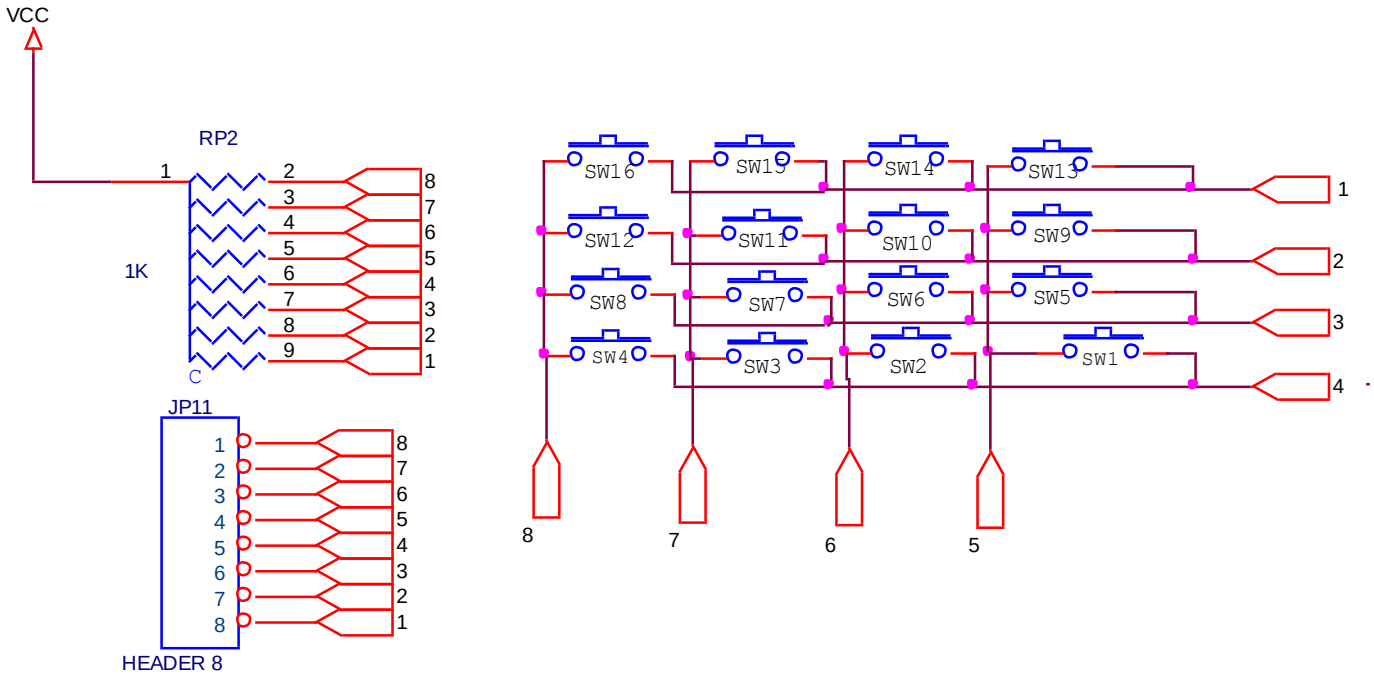
Sec-3] **Pulled down Push to On Switches – Eight Switches**



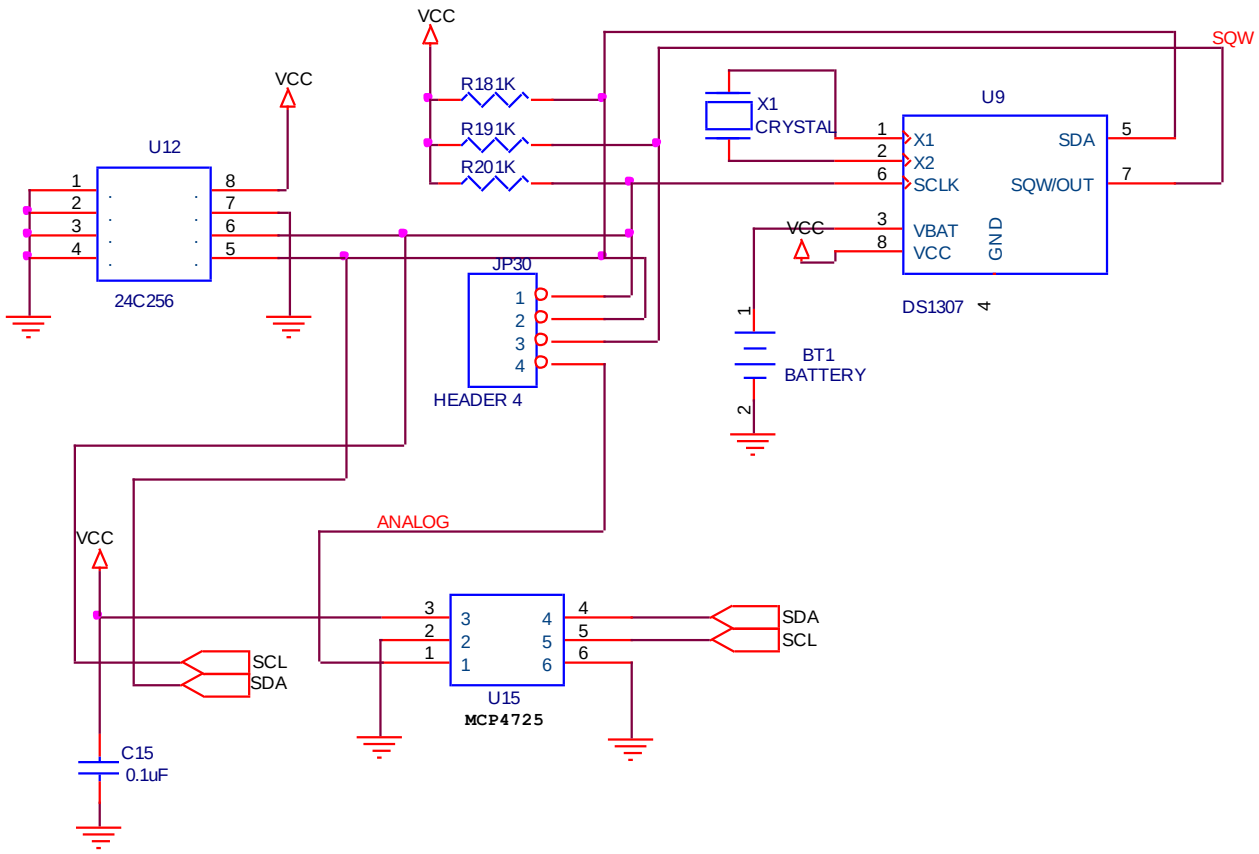
Sec-4-] **Pulled up Push to On Switches – Eight Switches**



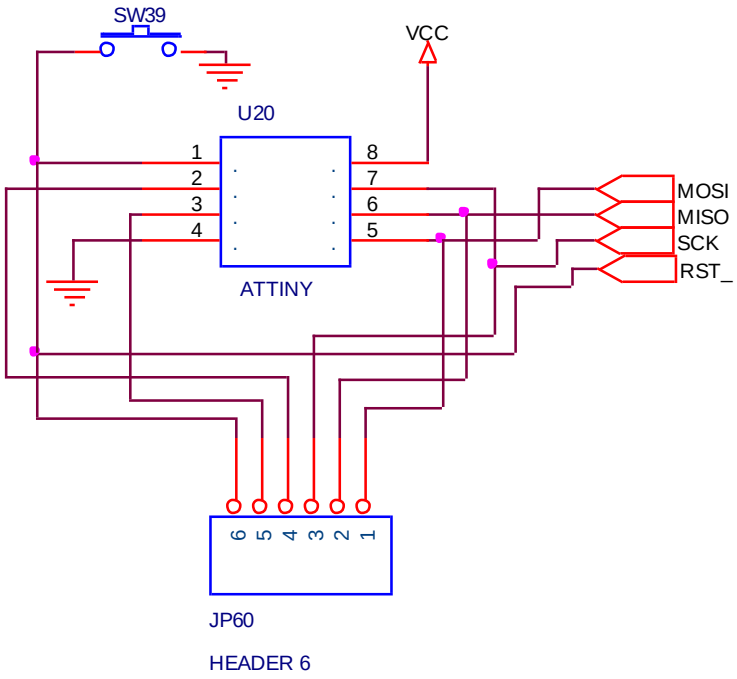
Sec-5] **Matrix keypad of 4x4 keys – Total Sixteen Keys**



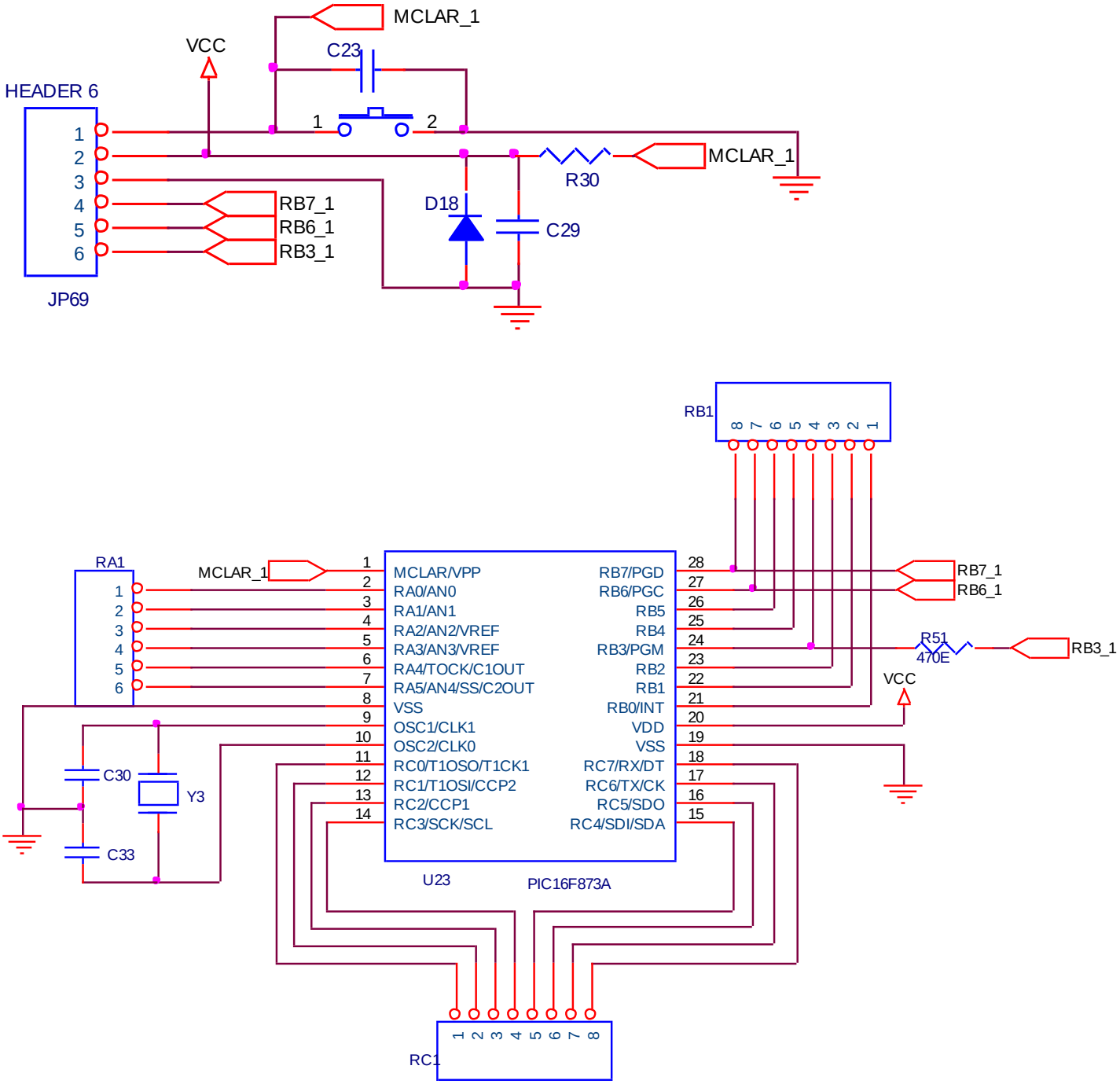
Sec-6] **I2C Based Real time clock, EEPROM & Digital to Analog Converter**



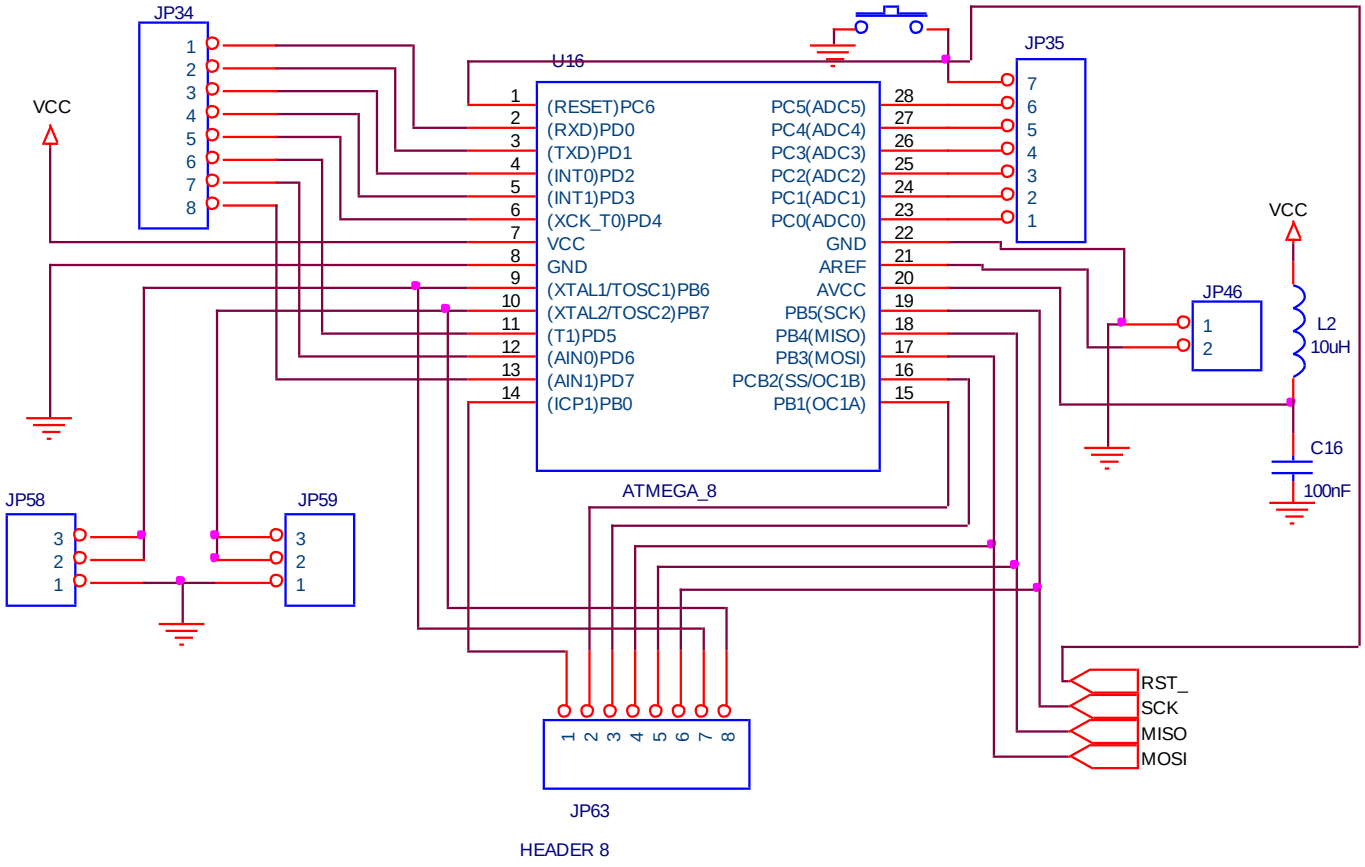
Sec-7] **Mount here ATTiny13 and other Pin compatible AVR Microcontrollers**



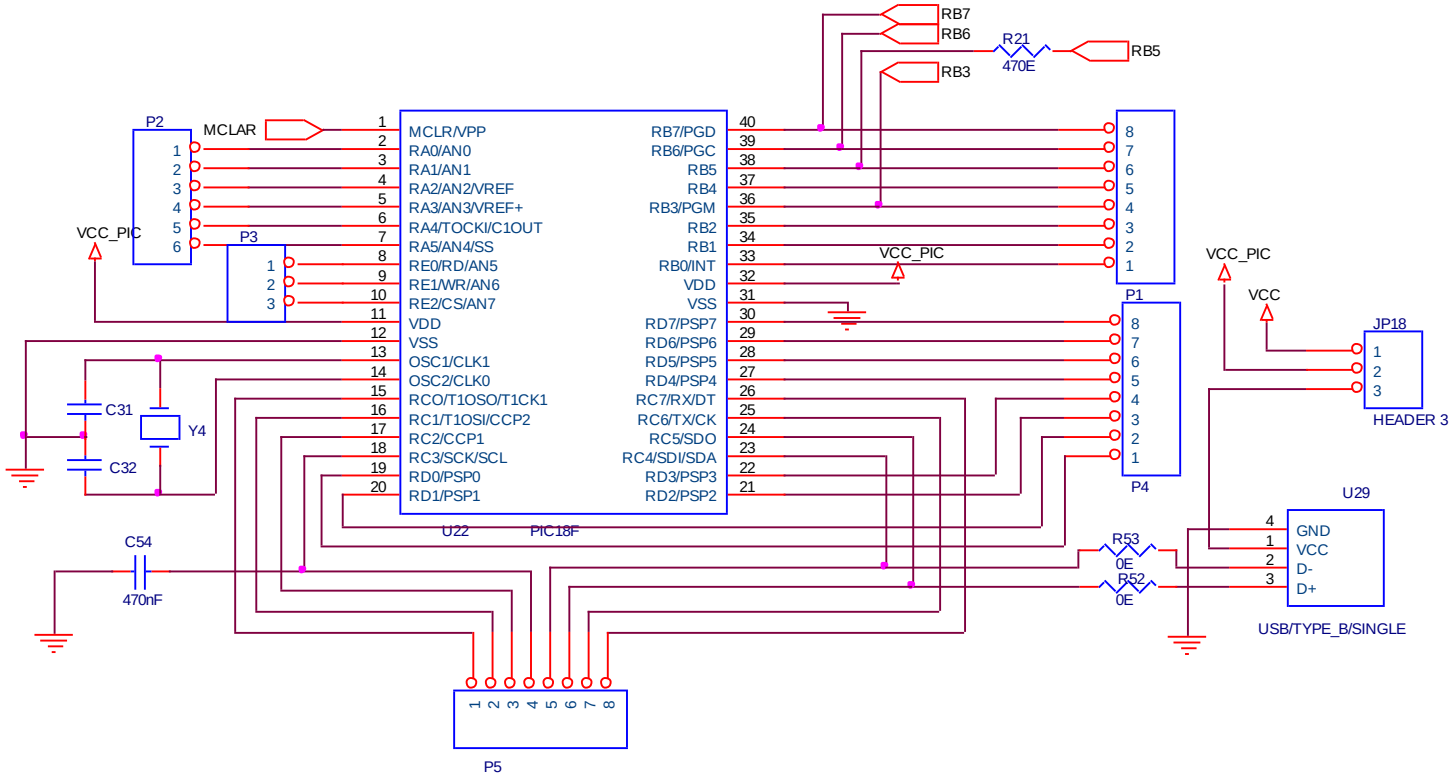
Sec-8] **Mount here PIC16F873A and other Pin compatible PIC Microcontrollers**

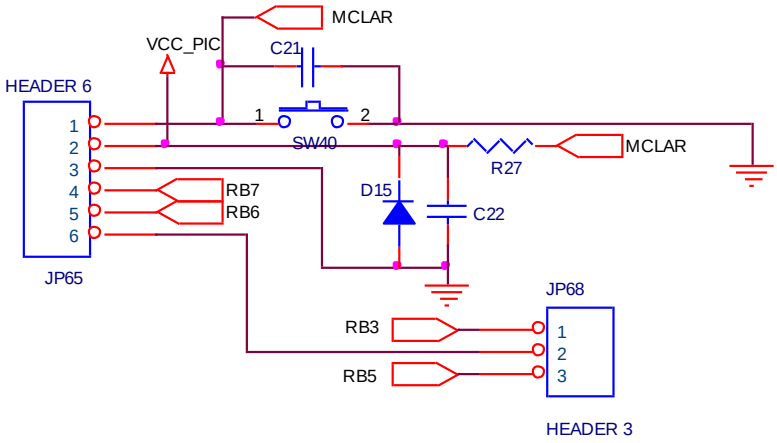


Sec-9] **Mount here ATmega8 / ATmega168 and other Pin compatible AVR Microcontrollers**

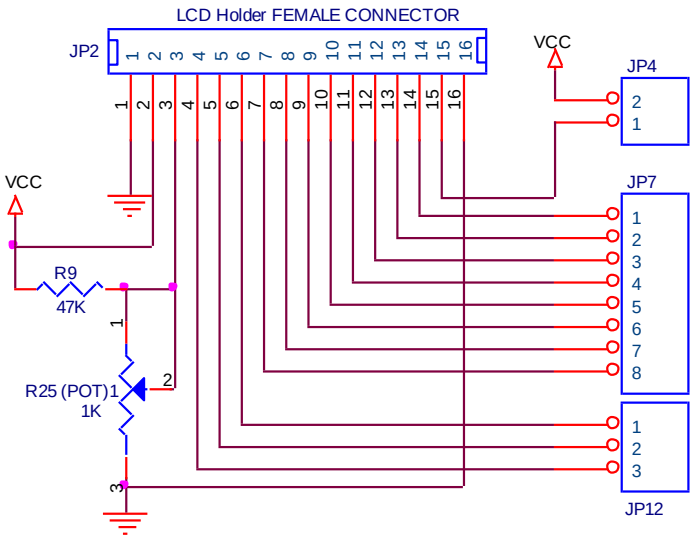


Sec-10] **Mount here PIC18F4550 / PIC18F452 and other Pin Compatible PIC Microcontrollers**

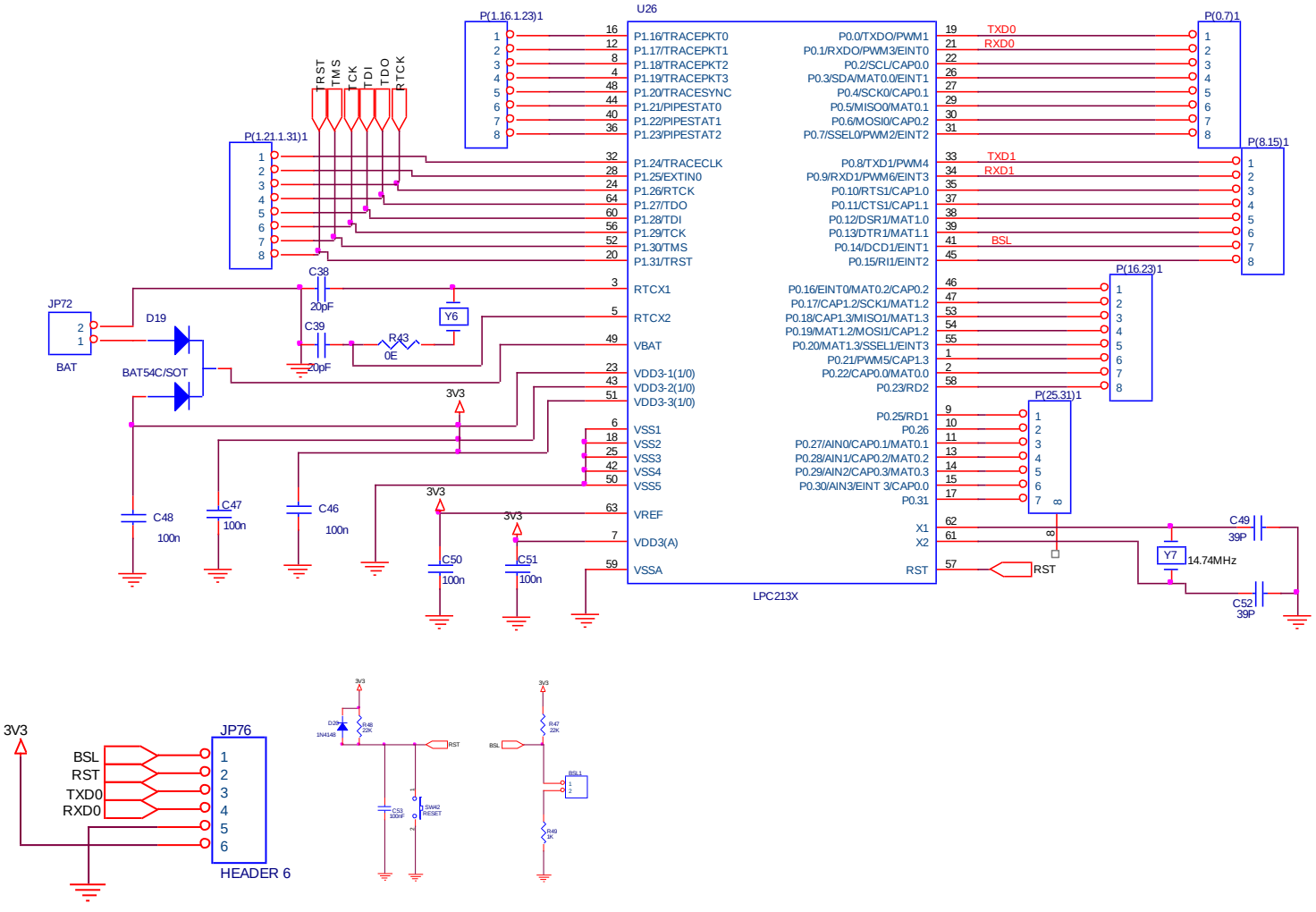




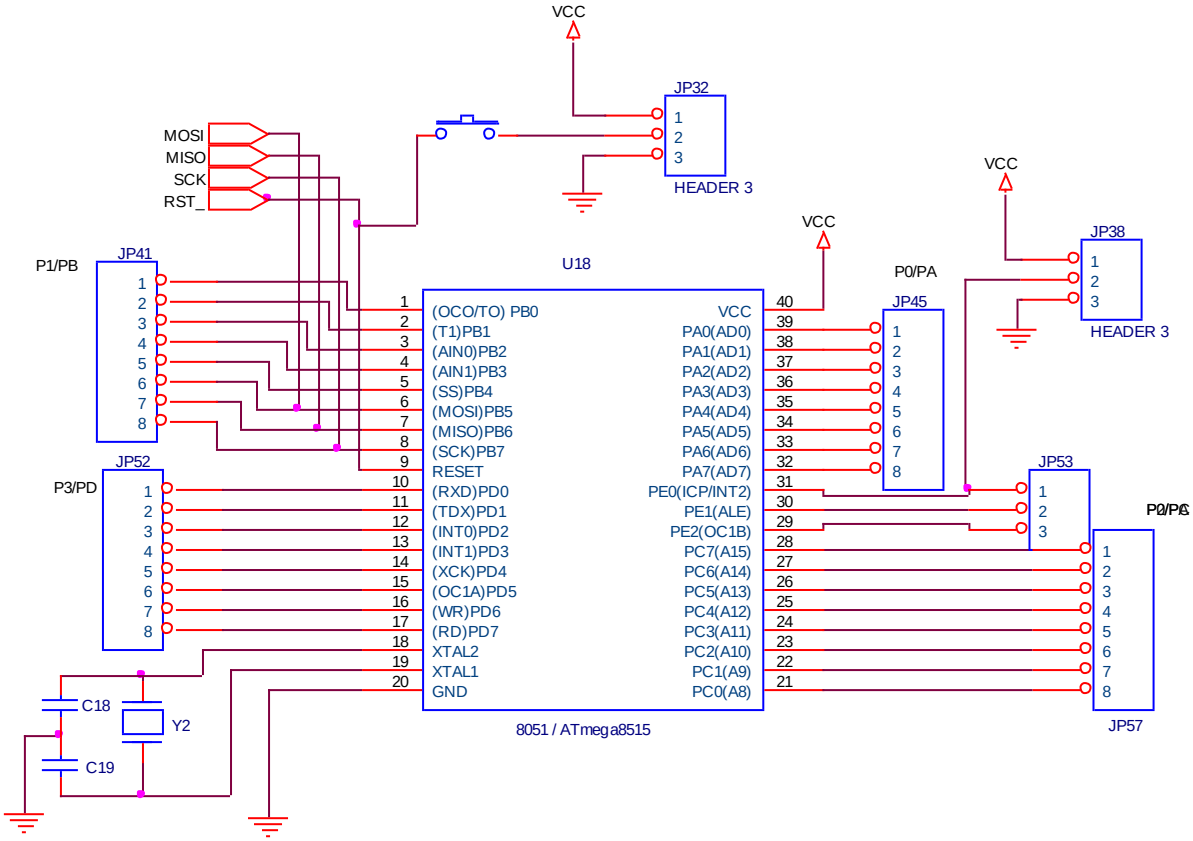
Sec-11] **16x2 LCD Interface. LCD is mounted on the board. Same place can be used for 20x4 LCD**



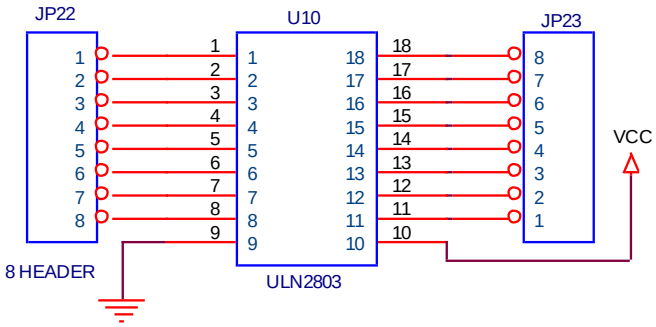
Sec-12] **Pre-soldered ARM7 - LPC2138 microcontroller**



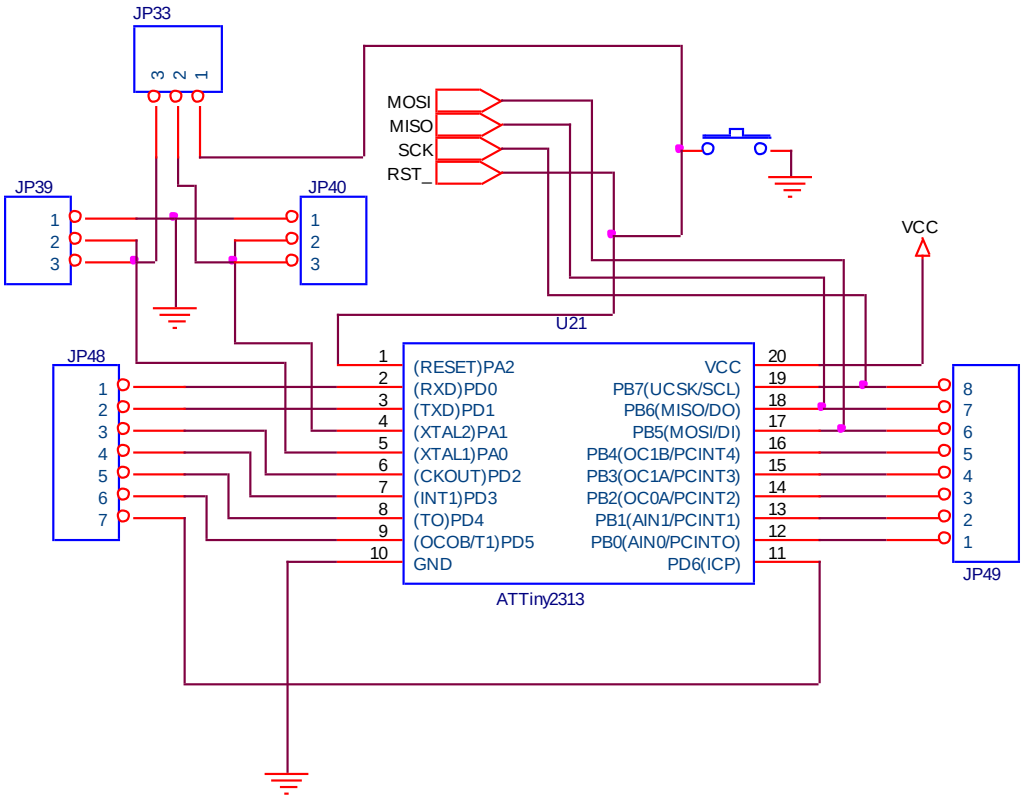
Sec-13] **Mount here P89V51RD2 (8051 family) microcontroller. This place can also be used for ATmega8515 AVR and its pin compatible microcontrollers. Reset pin Jumper position change required. Set it to Vcc when using 8051 family microcontroller. Set it to Gnd in case of AVR series.**



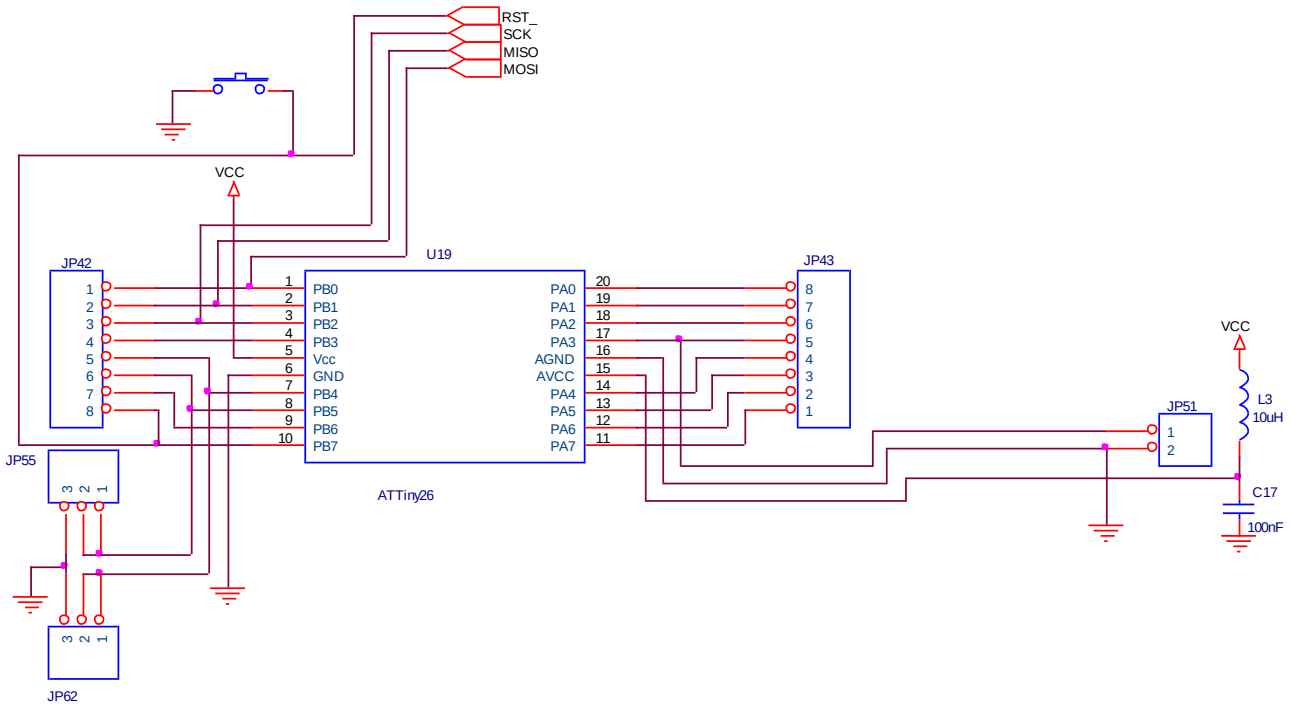
Sec-14] High Current Driver based on ULN2803 – Use it to drive Stepper motors, seven segment Displays.



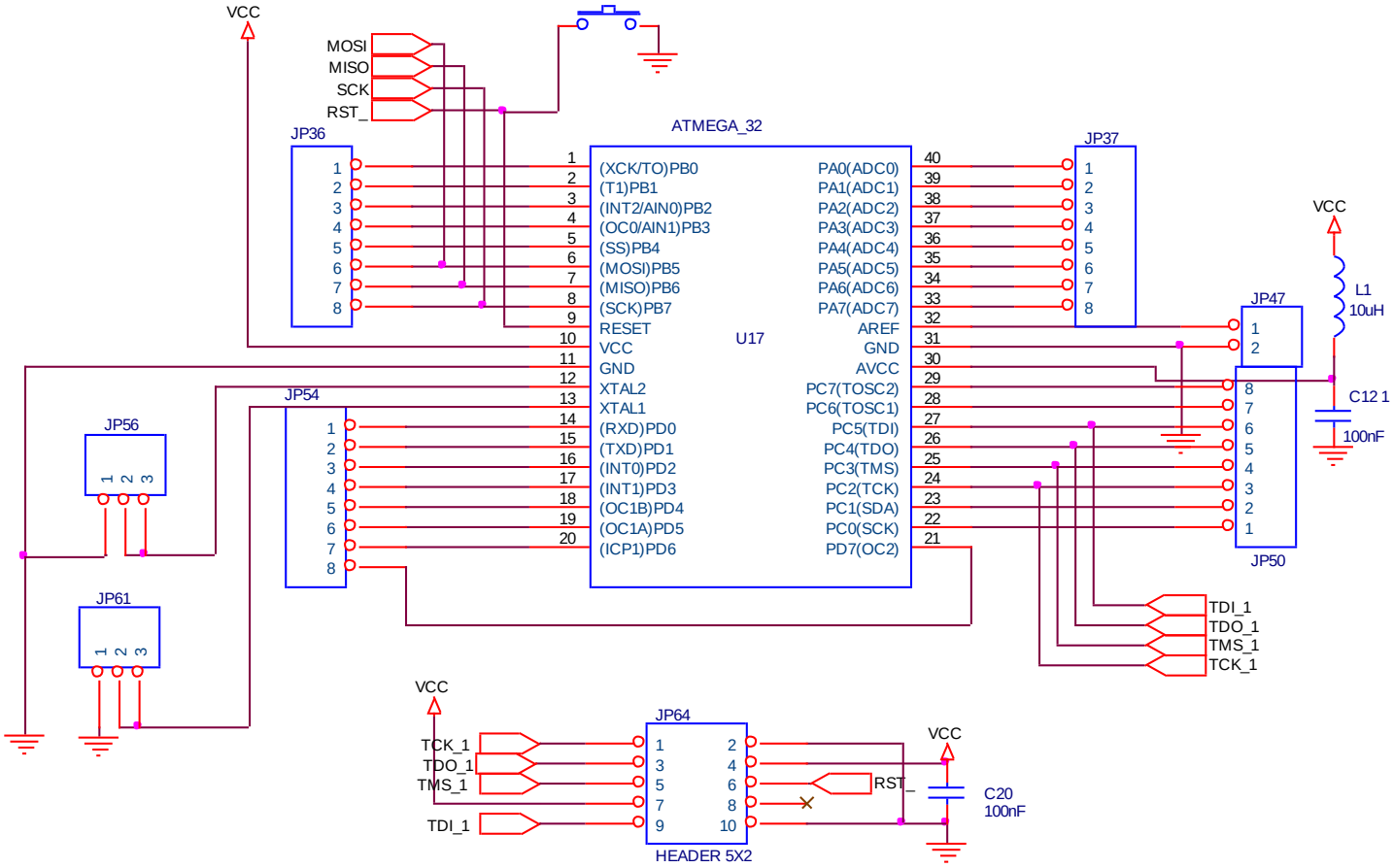
Sec-15] **Mount here ATTiny2313 and other pin compatible AVR Microcontrollers**



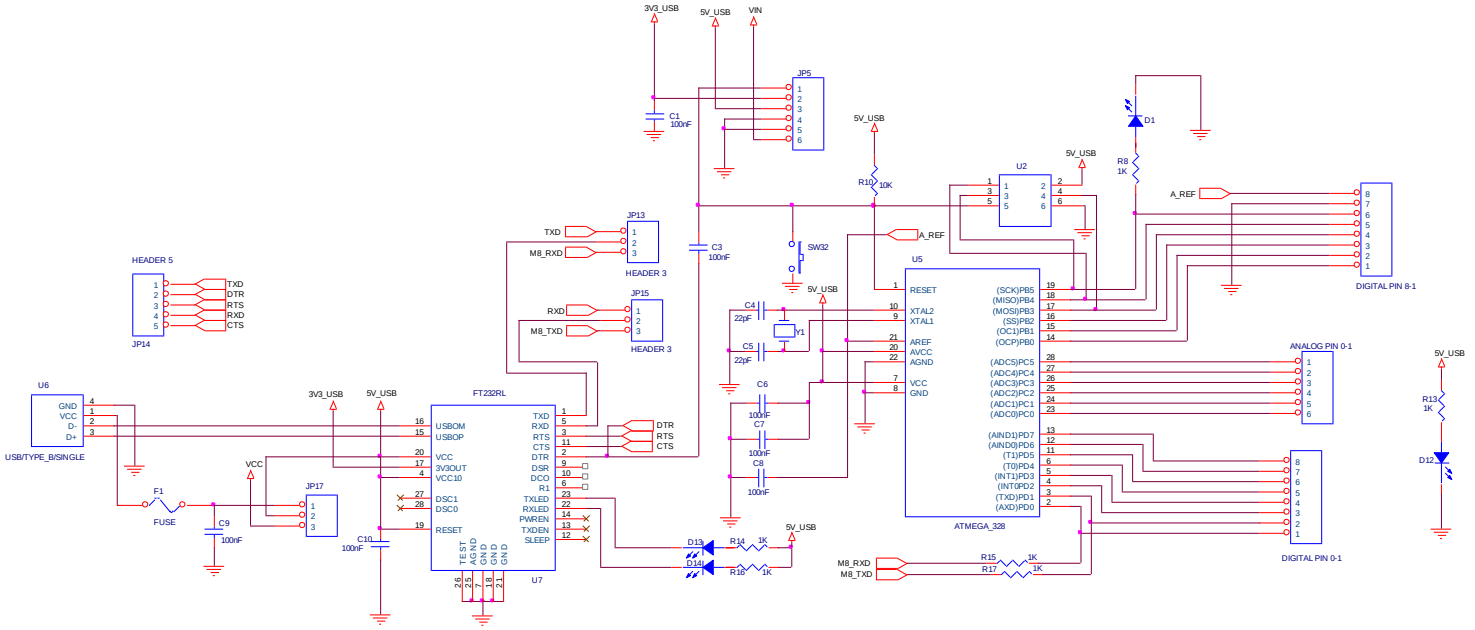
Sec-16] **Mount here ATTiny26 and other pin compatible AVR Microcontrollers**



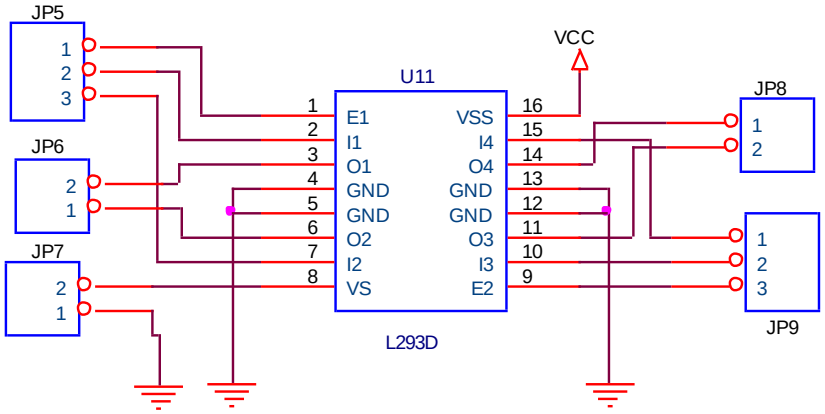
Sec-17] **Mount here ATmega16 / ATmega32 / ATmega8535 and other pin compatible AVR Microcontrollers**



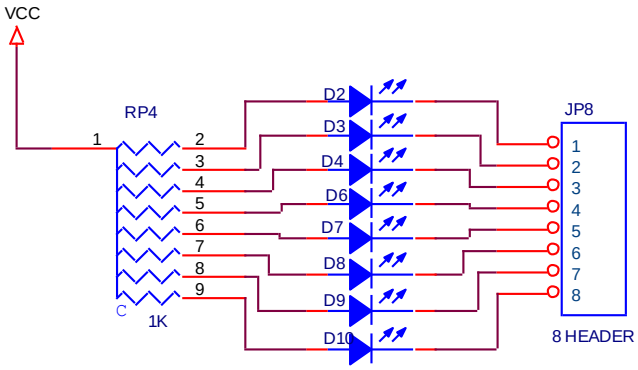
Sec-18] **Implementation of Arduino Duemilanove platform and Sec-23] USB to TTL converter interface – This is used for Arduino as well as a standalone USB to TTL interface. Jumper settings required.**



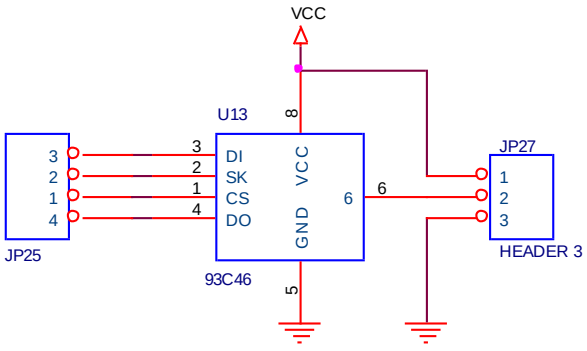
Sec-19] **DC Motor driver using L293D**



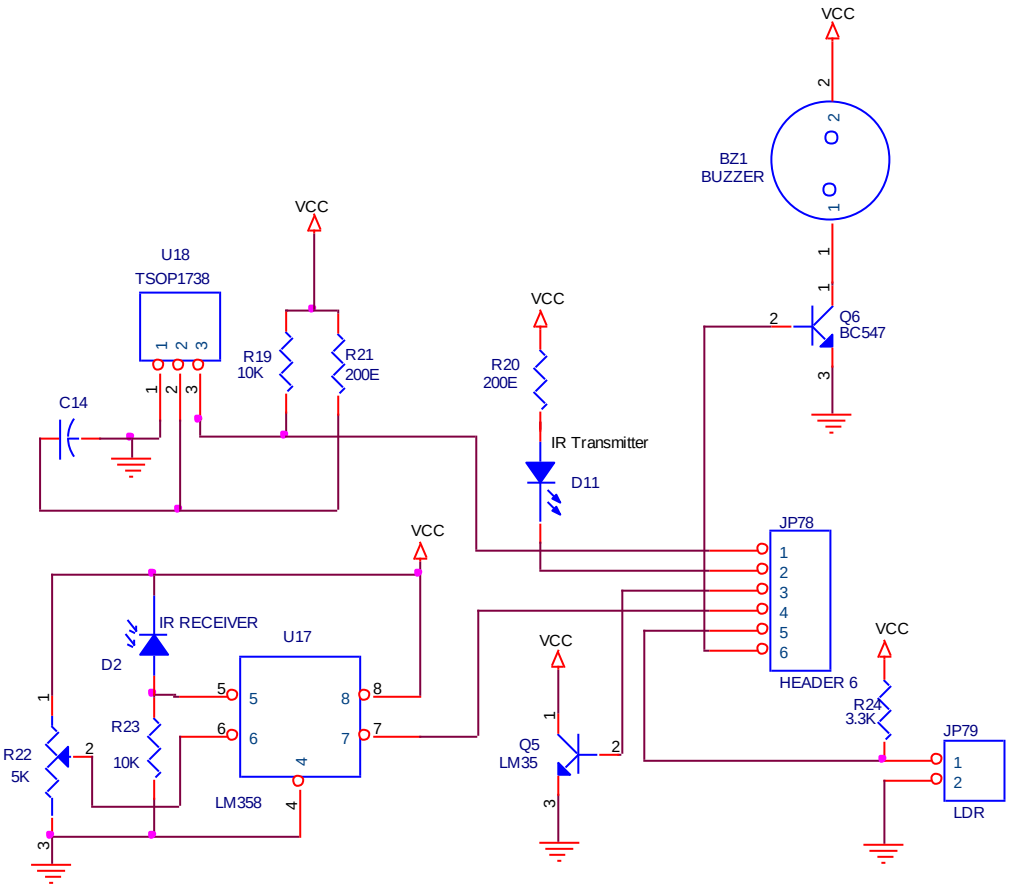
Sec-20] **A bank of 8 LEDs**



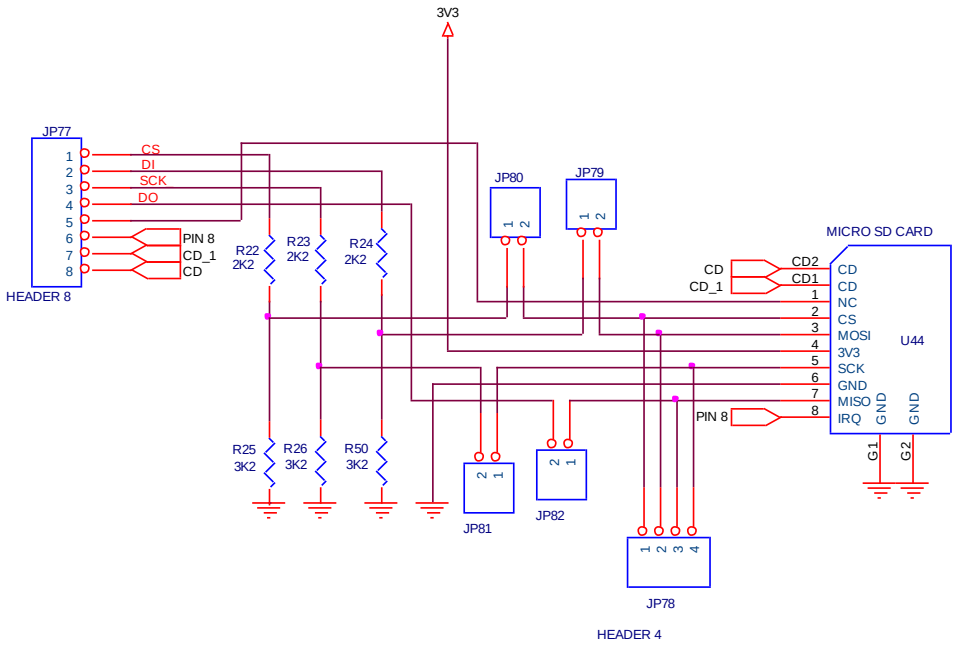
Sec-21] **SPI EEPROM AT93C46**



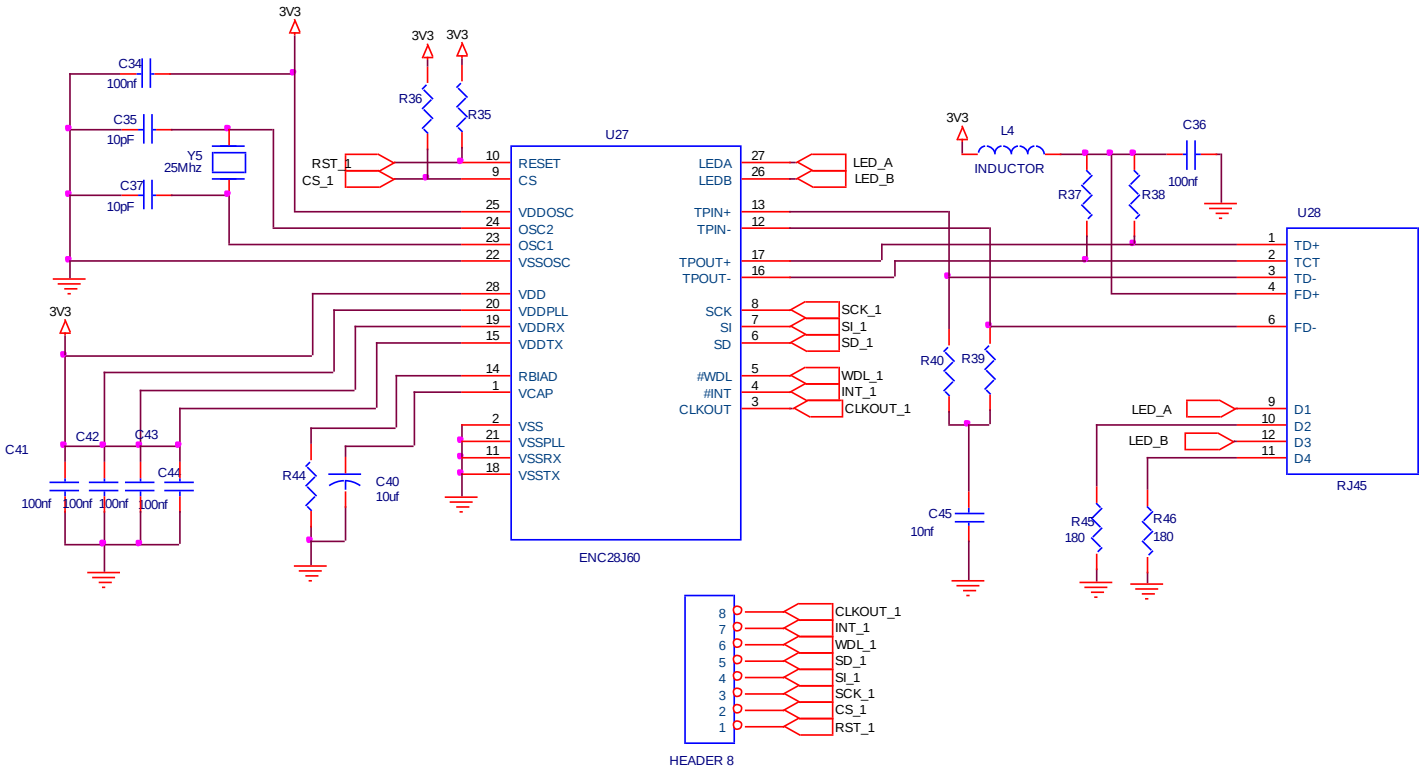
Sec-22] **Collection of sensors and other interfaces – Temperature sensor, Light Sensor, Infrared (IR) Sensor, 38KHz IR Receiver, IR Transmitter, Buzzer**



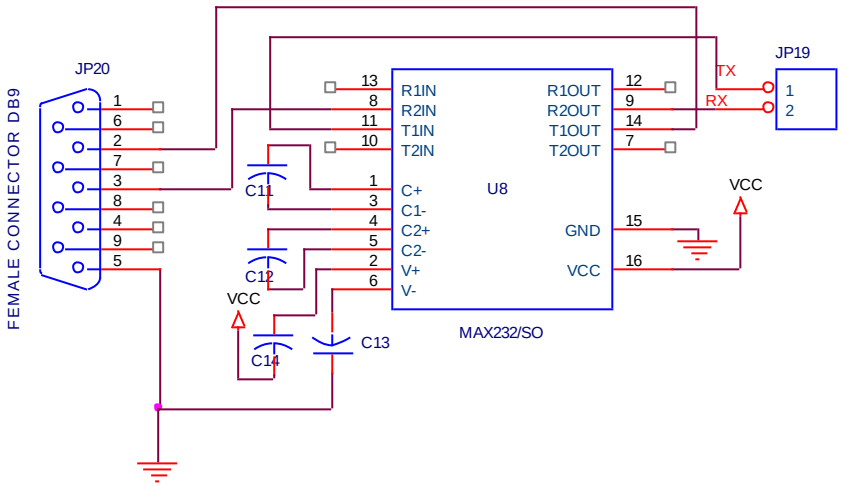
Sec-24] **Micro SD Card – Can be interfaced with 3.3V I/Os or with 5V I/Os. Jumper settings required.**



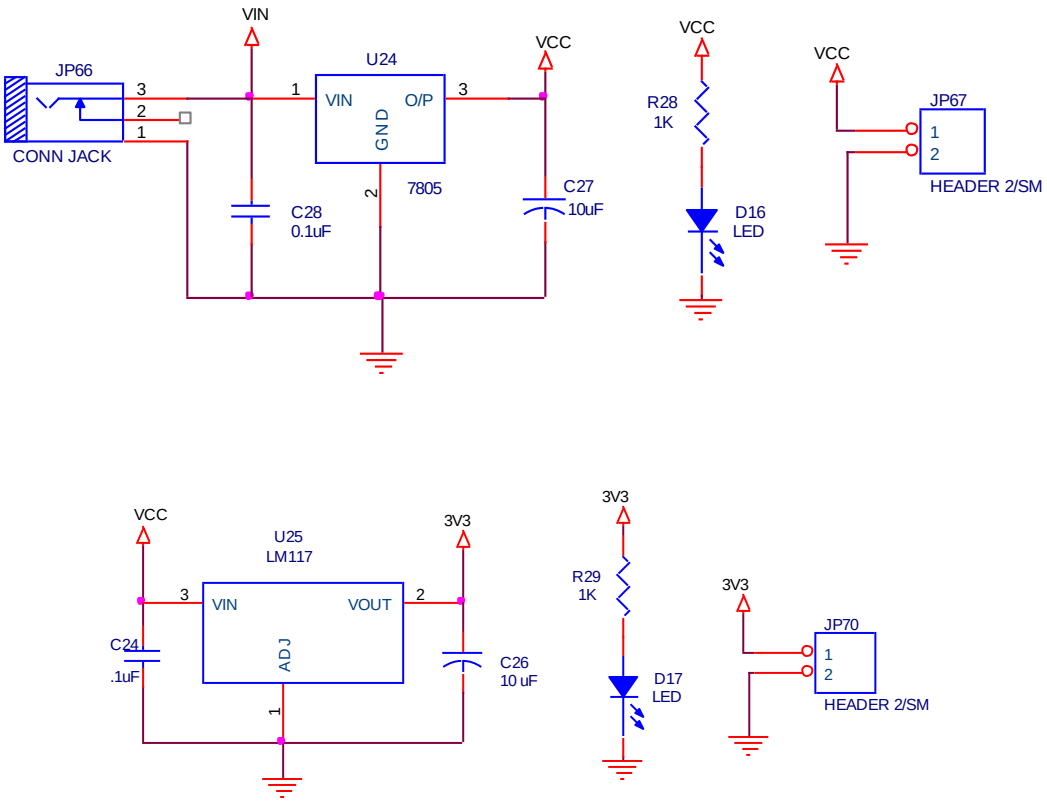
Sec-25] **ENC28J60 based Ethernet Interface. Use it to create web enabled applications.**



Sec-26] **RS232 to TTL Converter interface.**



Sec-27] **A Power supply section which includes 5V and 3.3V regulation. Wall mounting DC adapter which can source 500mA or higher at 9VDC is suitable. Exceeding input voltage may heat the onboard Voltage regulator ICs.**



11. Important information

1. EEDT6.0 product is designed for experiments and is not suitable to be used in life support and mission critical products.

2. EEDT6.0 requires 9VDC at 500mA or higher current source.

3. Always mount only 1 controller in any of the below listed sections:

    a. Sec – 7
    b. Sec – 9
    c. Sec – 13
    d. Sec – 15
    e. Sec – 16
    f. Sec – 17

    This is required because the ISP Programming port is common to all these sections. Thus there cannot be more than one microcontroller using the ISP Port.

    All these microcontrollers can be mounted and removed easily with fine tip screw driver or the IC extractor tool.

4. Always request support over email as it allows the technical team to answer it in more detail which is not possible over phone.

5. Manufactured by:

    Embedded Market
    205 Decision Tower
    Next To CityPride
    Satara Road
    Pune 411037 India
    Ph:+91 20 24228818
    Email for Support- support@embeddedmarket.com
    Email for Sales – sales@embeddedmarket.com
    Website – www.EmbeddedMarket.com